

Workflows Community Summit 2022

A Roadmap Revolution

High Performance
Data Management
and In Situ
Workflows

Specifications,
Standards, and
APIs

Workflows For
Continuum and
Cross-facility
Computing

AI Workflows

HPC and
Quantum
Workflows

FAIR Workflows

November 29 and 30, 2022

Disclaimer

The Workflows Community Summit was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

This report was prepared as an account of work sponsored by agencies of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

License

This report is made available under a Creative Commons Attribution 4.0 International Public license (<https://creativecommons.org/licenses/by/4.0>).

Preferred citation

R. Ferreira da Silva, R.M. Badia, V. Bala, D. Bard, T. Bremer, I. Buckley, S. Caino-Lores, K. Chard, C. Goble, S. Jha, D.S. Katz, D. Laney, M. Parashar, F. Suter, N. Tyler, T. Uram, I. Altintas, S. Andersson, W. Arndt, J. Aznar, J. Bader, B. Balis, C. Blanton, K.R. Braghetto, A. Brodutch, P. Brunk, H. Casanova, A. Cervera Lierta, J. Chigu, T. Coleman, N. Collier, I. Colonnelli, F. Coppens, M. Crusoe, W. Cunningham, B. de Paula Kinoshita, P. Di Tommaso, C. Doutriaux, M. Downton, W. Elwasif, B. Enders, C. Erdmann, T. Fahringer, L. Figueiredo, R. Filgueira, M. Foltin, A. Fouilloux, L. Gadelha, A. Gallo, A. Garcia, D. Garijo, R. Gerlach, R. Grant, S. Grayson, P. Grubel, J. Gustafsson, V. Hayot, O. Hernandez, M. Hilbrich, A. Justine, I. Laflotte, F. Lehmann, A. Luckow, J. Luettgau, K. Maheshwari, M. Matsuda, D. Medic, P. Mendygral, M. Michalewicz, J. Nonaka, M. Pawlik, L. Pottier, L. Pouchard, M. Pütz, S.K. Radha, L. Ramakrishnan, S. Ristov, P. Romano, D. Rosendo, M. Ruefenacht, K. Rycerz, N. Saurabh, V. Savchenko, M. Schulz, C. Simpson, R. Sirvent, T. Skluzacek, S. Soiland-Reyes, R. Souza, S.R. Sukumar, Z. Sun, A. Sussman, D. Thain, M. Titov, B. Tovar, A. Tripathy, M. Turilli, B. Tužnik, H. van Dam, A. Vivas, L. Ward, P. Widener, S.R. Wilkinson, J. Zawalska, M. Zulfiqar, “*Workflows Community Summit 2022: A Roadmap Revolution*”, Technical Report, ORNL/TM-2023/2885, March 2023, DOI: 10.5281/zenodo.7750670.

```
@techreport{wcs2022,
  author = {Ferreira da Silva, Rafael and Badia, Rosa M. and Bala, Venkat and Bard, Debbie
    and Bremer, Timo and Buckley, Ian and Caino-Lores, Silvina and Chard, Kyle and
    Goble, Carole and Jha, Shantenu and Katz, Daniel S. and Laney, Daniel and
    Parashar, Manish and Suter, Frederic and Tyler, Nick and Uram, Thomas and
    Altintas, Ilkay and Andersson, Stefan and Arndt, William and Aznar, Juan and
    Bader, Jonathan and Balis, Bartosz and Blanton, Chris and Braghetto, Kelly Rosa
    and Brodutch, Aharon and Brunk, Paul and Casanova, Henri and Cervera Lierta, Alba
    and Chigu, Justin and Coleman, Taina and Collier, Nick and Colonnelli, Iacopo and
    Coppens, Frederik and Crusoe, Michael and Cunningham, Will and de Paula Kinoshita,
    Bruno and Di Tommaso, Paolo and Doutriaux, Charles and Downton, Matthew and Elwasif,
    Wael and Enders, Bjoern and Erdmann, Chris and Fahringer, Thomas and Figueiredo,
    Ludmilla and Filgueira, Rosa and Foltin, Martin and Fouilloux, Anne and Gadelha,
    Luiz and Gallo, Andy and Garcia, Artur and Garijo, Daniel and Gerlach, Roman and
    Grant, Ryan and Grayson, Samuel and Grubel, Patricia and Gustafsson, Johan and
    Hayot, Valerie and Hernandez, Oscar and Hilbrich, Marcus and Justine, Annmary and
    Laflotte, Ian, and Lehmann, Fabian and Luckow, Andre and Luettgau, Jakob and
    Maheshwari, Ketan and Matsuda, Motohiko and Medic, Doriana and Mendygral, Pete and
    Michalewicz, Marek and Nonaka, Jorji and Pawlik, Maciej and Pottier, Loic and
    Pouchard, Line and Putz, Mathias and Radha, Santosh Kumar and Ramakrishnan, Lavanya
    and Ristov, Sashko and Romano, Paul and Rosendo, Daniel and Ruefenacht, Martin and
    Rycerz, Katarzyna and Saurabh, Nishant and Savchenko, Volodymyr and Schulz, Martin
    and Simpson, Christine and Sirvent, Raul and Skluzacek, Tyler and Soiland-Reyes,
    Stian and Souza, Renan and Sukumar, Sreenivas Rangan and Sun, Ziheng and Sussman,
    Alan and Thain, Douglas and Titov, Mikhail and Tovar, Benjamin and Tripathy, Aalap
    and Turilli, Matteo and Tuznik, Bartosz and van Dam, Hubertus and Vivas, Aurelio
    and Ward, Logan and Widener, Patrick and Wilkinson, Sean R. and Zawalska, Justyna
    and Zulfiqar, Mahnoor},
  title = {{Workflows Community Summit 2022: A Roadmap Revolution}},
  month = {March},
  year = {2023},
  publisher = {Zenodo},
  number = {ORNL/TM-2023/2885},
  doi = {10.5281/zenodo.7750670},
  institution = {Oak Ridge National Laboratory}
}
```

Table of Contents

1	Introduction	5
1.1	Summit Organization	5
1.2	Summit Structure and Activities	5
2	An Overview of the 2021 Community Roadmap for Scientific Workflows Research and Development	8
3	Specifications, Standards, and APIs	9
3.1	Workflow Stack	9
3.2	Standardization	10
4	AI Workflows	11
4.1	AI Workflows Characteristics	11
4.2	AI Workflows Categories and Motifs	12
5	High Performance Data Management and In Situ Workflows	13
6	HPC and Quantum Workflows	15
6.1	State-of-the-art and Challenges	16
7	FAIR Computational Workflows	16
7.1	FAIR Workflow Repositories and Registries	17
7.2	Building FAIR into Workflow Management Systems	17
8	Workflows for Continuum and Cross-Facility Computing	18
8.1	Characteristics and Challenges	19
	References	21
	Appendix A: Participants and Contributors	28
	Appendix B: Agenda	31

1 Introduction

Scientific workflows have become integral tools in broad scientific computing use cases [1]. Science discovery is increasingly dependent on workflows to orchestrate large and complex scientific experiments that range from the execution of a cloud-based data preprocessing pipeline to multi-facility instrument-to-edge-to-HPC computational workflows [2, 3]. Given the changing landscape of scientific computing (often referred to as a computing continuum [4]) and the evolving needs of emerging scientific applications, it is paramount that the development of novel scientific workflows and system functionalities seek to increase the efficiency, resilience, and pervasiveness of existing systems and applications. Specifically, the proliferation of machine learning/artificial intelligence (ML/AI) workflows, need for processing large-scale datasets produced by instruments at the edge, intensification of near real-time data processing, support for long-term experiment campaigns, and emergence of quantum computing as an adjunct to HPC, have significantly changed the functional and operational requirements of workflow systems. Workflow systems now need to, for example, support data streams from the edge-to-cloud-to-HPC [5], enable the management of many small-sized files [6], allow data reduction while ensuring high accuracy [7], orchestrate distributed services (workflows, instruments, data movement, provenance, publication, etc.) across computing and user facilities [8], among others. Further, to accelerate science, it is also necessary that these systems implement specifications/standards and APIs for seamless (horizontal and vertical) integration between systems and applications [9, 10], as well as enable the publication of workflows and their associated products according to the FAIR principles [11].

1.1 Summit Organization

This document reports on discussions and findings from the 2022 international edition of the “Workflows Community Summit” that took place on November 29 and 30, 2022 [12]. The two-day summit included 106 participants (Figure 1), from a group of international researchers and developers (Australia, Austria, Belgium, Brazil, Canada, Colombia, Egypt, France, Germany, Italy, Japan, Netherlands, Norway, Poland, Spain, Switzerland, United Kingdom, United States) from distinct workflow management systems and users, and representatives from funding agencies and industry.

The summit was organized by lead members of the Workflows Community Initiative (WCI) [13], a volunteer effort that brings the workflows community together (users, developers, researchers, and facilities) to provide community resources and capabilities to enable scientists and workflow systems developers to discover software products, related efforts, events, technical reports, etc. and engage in community-wide efforts to tackle workflows grand challenges. These members efforts have been supported by distinct research projects that contributed to the organization of the event, including ExaWorks [10], WorkflowHub [14], eFlows4HPC [15], and Covalent [16].

1.2 Summit Structure and Activities

Based on the outcomes of the 2021 summits [17–19], the community developed a roadmap for workflows research and development [6] (summarized in Section 2). From this roadmap, we have identified six cross-cutting research/technical topics for the 2022 edition of the summit, each of which was the object of a focused discussion led by a volunteer community member (Table 1). (Note that the first three topics were discussed

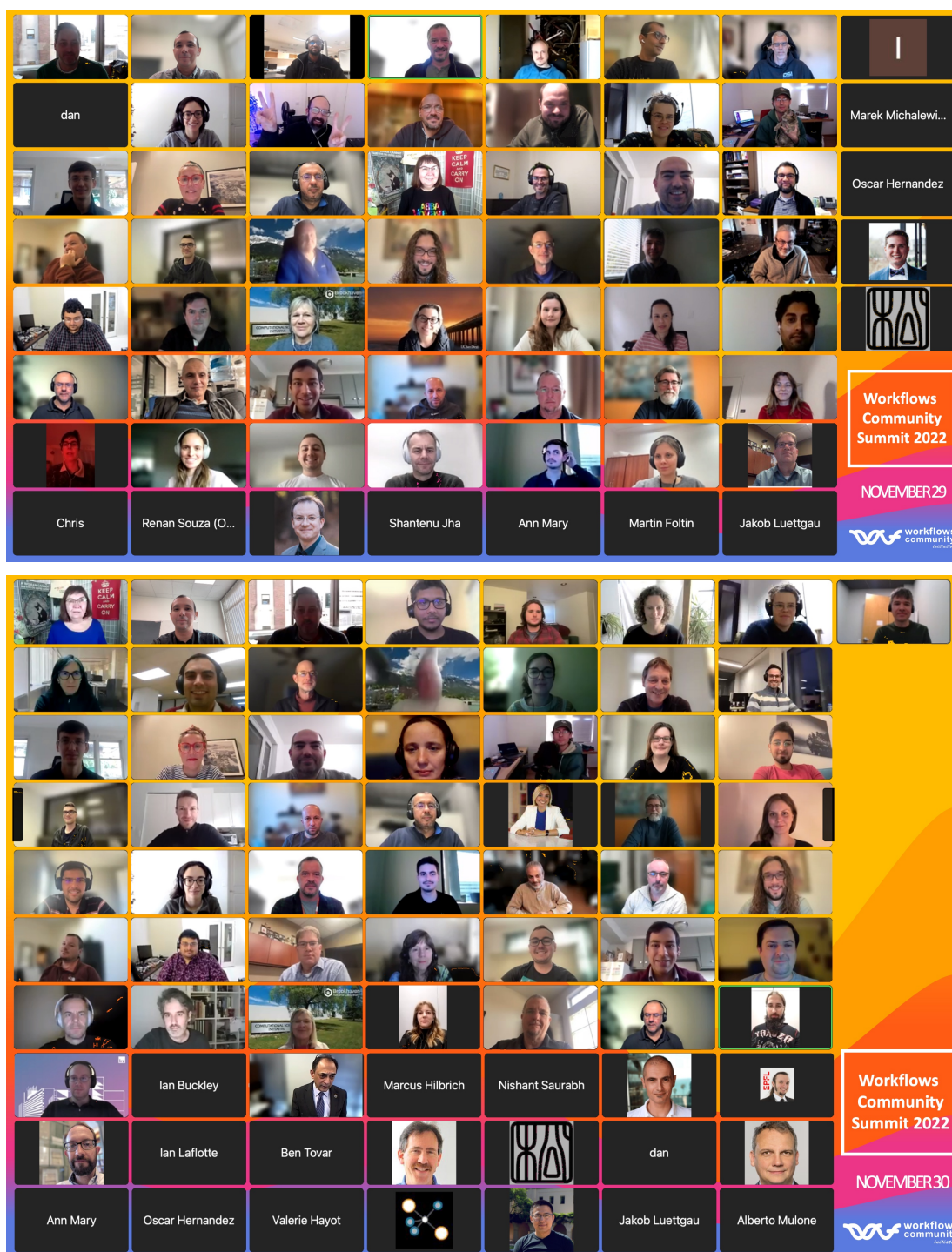


Figure 1: Screenshot of the 2022 edition of the Workflows Community Summit participants. (The event was held virtually via Zoom on November 29 and 30, 2022.)

Topic	Discussion Co-Leaders
Topic 1: Specifications, standards, and APIs	Daniel Laney (LLNL), Kyle Chard (UChicago/ANL)
Topic 2: AI workflows	Shantenu Jha (BNL), Timo Bremer (LLNL)
Topic 3: High performance data management and in situ workflows	Silvina Caino-Lores (UTK), Frédéric Suter (ORNL)
Topic 4: HPC and Quantum workflows	Rosa M. Badia (BSC), Ian Buckley (Agnostiq), Venkat Bala (Agnostiq)
Topic 5: FAIR computational workflows	Carole Goble (UManchester), Daniel S. Katz (UIUC)
Topic 6: Workflows for continuum and cross-facility computing	Debbie Bard (NERSC), Tom Uram (ANL), Nick Tyler (NERSC)

Table 1: Workflows Community Summit topics and discussion co-leaders.

in the first day of the event, and the latter three topics in the second day.) For each of the technical topics shown in Table 1, the co-leaders gave a plenary 10-minute lightning talk followed by focused discussions in breakout sessions. The goal of these sessions was to (i) review the previous challenges and proposed solutions; and (ii) identify crucial gaps and potential short- and long-term solutions for enabling emerging and new workflow applications given the rapid evolution of the computing continuum paradigm. The lead then reported on the outcome of the discussion in plenary sessions, after which final remarks were given by the organizers and the summit was adjourned. Additionally, the summit included an invited talk from **Prof. Manish Parashar** (University of Utah), a renowned researcher in the field of Scientific Computing, entitled “A Translational Perspective on End-to-End Workflows.” The cross-cutting theme of Parashar’s talk was discussed throughout all breakout sessions.

All presentations and videos can be found in the summit website (<https://workflows.community/summits/2022>), and videos can be watched from WCI’s YouTube channel (https://www.youtube.com/playlist?list=PLAtmuqHEXrVO-1-mACm1K_4mfmG6yLnpN).

2 An Overview of the 2021 Community Roadmap for Scientific Workflows Research and Development

The overarching goal of the 2021 Workflows Community Summits was to (i) develop a view of the state of the art, (ii) identify key research challenges, (iii) articulate a vision for potential activities, and (iv) explore technical approaches for realizing (part of) this vision. The summits gathered lead researchers and developers from around the world, and spanning distinct workflow systems and user communities. We synthesized the discussions and outcomes of the previous summits and findings into a community roadmap [6] that presents a consolidated view of the state of the art, challenges, and potential activities. Table 2 presents, in the form of top-level themes, a summary of those challenges and targeted community activities. Table 3 summarizes a

Theme	Challenges	Community Activities
FAIR Computational Workflows	- FAIR principles for computational workflows that consider the complex lifecycle from specification to execution and data products - Metrics to measure the “FAIRness” of a workflow - Principles, policies, and best practices	- Review prior and current efforts for FAIR data and software with respect to workflows, and outline principles for FAIR workflows - Define recommendations for FAIR workflow developers and systems - Automate FAIRness in workflows by recording necessary provenance data
AI Workflows	- Support for heterogeneous compute resources and fine-grained data management features, versioning, and data provenance capabilities - Capabilities for enabling workflow steering and dynamic workflows - Integration of ML frameworks into the current HPC landscape	- Develop comprehensive use cases for sample problems with representative workflow structures and data types - Define a process for characterizing the challenges for enabling AI workflows - Develop AI workflows as a way to benchmark HPC systems
Exascale Challenges and Beyond	- Resource allocation policies and schedulers are not designed for workflow-aware abstractions, thus users tend to use an ill-fitted job abstraction - Unfavorable design of resource descriptions and mechanisms for workflow users/systems, and lack of fault-tolerance and fault-recovery solutions	- Develop documentation in the form of workflow templates/recipes/miniapps for execution on high-end HPC systems - Specify benchmark workflows for exascale execution - Include workflow requirements as part of the machine procurement process
APIs, Reuse, Interoperability, and Standards	- Workflow systems differ by design, thus interoperability at some layers is likely to be more impactful than others - Workflow standards are typically developed by a subset of the community - Quantifying the value of common representations of workflows is not trivial	- Identify differences and commonalities between different systems - Identify and characterize domain-specific efforts, identify workflow patterns, and develop case-studies of business process workflows and serverless workflow systems
Training and Education	- Many workflow systems have high barriers to entry and lack training materials - Homegrown workflow solutions and constraints can prevent users from reproducing their functionality on workflow systems developed by others - Unawareness of the workflow technological and conceptual landscape	- Identify basic sample workflow patterns, develop a community workflow knowledge-base, and look at current research on technology adoption - Include workflow terminology and concepts in university curricula and software carpentry efforts
Building a Workflows Community	- Diverse definitions of a “workflows community” - Remedy the inability to link developers and users to bridge translational gaps - Pathways for participation in a network of researchers, developers, and users	- Establish a common knowledge-base for workflow technology - Establish a <i>Workflow Guild</i>: an organization focused on interaction and relationships, providing self-support between workflow developers and their systems

Table 2: Summary of workflows research and development challenges and proposed community activities identified during the 2021 summits [6].

Thrust	Roadmap Milestones
Definition of common workflow patterns and benchmarks	• Define small sets of workflow patterns and benchmark deliverables, and implement them using a selected set of workflow systems • Investigate automatic generation of patterns and configurable benchmarks (e.g., to enable weak and strong scaling experiments) • Establish or leverage a centralized repository to host and curate patterns and benchmarks
Identifying paths toward interoperability of workflow systems	• Define interoperability for different roles, develop a horizontal interoperability (i.e., making interoperable components), and establish a requirements document per abstraction layer • Develop real-world workflow benchmarks, use cases for interoperability, and common APIs that represent workflow library components • Establish a workflow systems developer community
Improving workflow systems' interface with legacy and emerging HPC software and hardware stacks	• Document a machine-readable description of key properties of widely used sites, and remote authentication needs from the workflow perspective • Identify new workflow patterns (e.g., motivated by AI workflows), attain portability across heterogeneous hardware, and develop a registry of execution environment information • Organize a community event involving workflow system developers, end users, authentication technology providers, and facility operators

Table 3: Summary of technical roadmap milestones per research and development thrust proposed during the 2021 summits [6].

proposed community roadmap with technical approaches.

In the 2022 edition of the Workflows Community Summit, we revisited this roadmap, discussed the relevance of the challenges and proposed milestones given the current landscape of workflow applications needs and emerging infrastructures. In the following sections, we summarize the outcomes of these discussions.

3 Specifications, Standards, and APIs

Defining reference specifications, standards, and APIs is paramount to enable interoperability across the workflow software stack; for capturing end-to-end workflow provenance to facilitate user understanding of the scientific output; for enabling composability of underlying workflow system components to enable seamless exchange of functional modules; among others. The workflows community has tackled this problem in many ways ranging from solutions that provide reference implementations of standards and/or specifications [10,20] to define common patterns/motifs [21,22]. In this breakout session, participants focused on identifying the challenges impeding adoption of current standards, specifications, best practices, and APIs developed by the community. Below, we summarize the key outcomes of these discussions, highlighting exemplar efforts, and outline proposed actionable solutions to be tackled by the community.

3.1 Workflow Stack

Despite the vast efforts for providing reference standard implementations and identifying common patterns, defining the levels of the workflow stack which are amenable to sharing technologies (and potentially APIs) is still an open question and potentially one of the critical aspects hindering wide adoption of proposed standards and specifications. For instance, there are several ways to define a workflow stack: (i) consider different types of workflows based on their resource requirements (e.g., in situ, intra-site, inter-site); (ii) con-

sider a vertical hierarchical stack starting from the workflow down to computational jobs and individual tasks; (iii) consider the kind of stakeholder and their needs, etc.

Given the above, an overarching question is how do these different workflow stacks map to different types of workflow systems? Examples of such aspects include resource authentication/authorization and allocation; the granularity of jobs definition/submission: fine-grained (functions) vs. coarse-grained (long running jobs); data management (from/to external data stores, within the workflow system, and within the tasks/serverless functions); workflow specification, language, and framework; provenance and error reporting; visualization; and application deployment (native installation vs. containerization).

The Global Alliance for Genomics & Health (GA4GH) [23] has targeted the development of API standards (and implementations) that make it easier to “send the algorithms to the data” on a cloud-based environment. Their workflow stack includes four definitions of standard APIs for interacting with a data repository, tool registry, task execution, and workflow execution services [24]. The Pulsar-Network project [25] provides a distributed, uniform job execution system across European data centers, in which the execution environment is detached from the Galaxy framework. NERSC’s SuperFacility API [26] exposes characteristics of the computational facility’s resources that could be leveraged by decision making systems. One limitation of these approaches is the need for describing every single component of the stack. When considering heterogeneous platforms (Cloud, HPC, Edge, or the emerging quantum computing environments) the definition of such standards may become overly specialized to a specific platform or service [21]. The Common Workflow Scheduler [27] API allows workflow systems to exchange scheduling information with resource managers, making them “workflow-aware” and capable of scheduling tasks based on workflow dependencies and optimization goals. This approach eliminates the need for workflow systems to implement resource manager-specific logic and simplifies the separation of scheduling responsibilities between workflow systems and resource managers. Currently, this API is available as a plugin for Nextflow and Kubernetes.

Recommendations. There is an imminent need for a description of a common workflow stack that might need to be represented by different sets of characterizations depending on the execution environment, however each described using common patterns identified across distinct environments. Once the fundamental workflow stack is defined and understood, the community could focus on the development of standard APIs as the result of a long-term objective.

3.2 Standardization

Attaining standardization is a foremost requirement to achieve interoperability, portability, and reuse of workflow components and applications. However, full standardization must be accomplished at different levels of the workflow stack. For instance, user-facing standards consider workflow descriptions and workflow input objects, while resource/infrastructure standards target the description of services that can be leveraged by, for example, workflow systems. In addition to standards, shared libraries and designs are also essential for interchanging components and services at horizontal levels of the workflow stack.

One of the challenging aspects of defining standards is the lack of a common terminology that is widely adopted by the community. Unfortunately, the workflows literature suffers from an absence of a common vocabulary that defines terms and describes the components of a workflow. Recent efforts to provide a consolidated view [28] or reuse [29] of specification languages for a subset of workflow systems has underlined the discrepancy of term definitions. (Dissimilarities can be even identified at the simple definition of tasks and jobs across workflow systems.) Additionally, it is also necessary to describe the semantics of the

components and of the data, as well as the variety of systems on which workflows are executed.

Standardization is also required at the metadata level (user-defined metadata prior to data generation, ad-hoc metadata to annotate a dataset [30], and workflow-based metadata) [31], which is key for enabling provenance. The Workflow Run RO-Crate working group [32] is defining profiles for capturing the provenance of an execution of a computational workflow based on different levels of granularity: process (execution of one or more tools that contribute to the same computation), workflow (coordinated execution of the tools driven by a workflow system), and provenance (internal details of each step of the workflow). These profiles are based on the CWLProv profile [33], which are organized based on CWL's workflow mode. Although CWLProv is not a standard such as W3C PROV, it attempts to fill the gaps when translating the latter's definitions to workflow specifications (e.g., traditional standards are hard to adopt into HPC due to their origins in other domains). A remaining open question is how to increase adoption of these standards to other workflow systems, and how to capture specifics of heterogeneous systems without hiding the level of provenance the user cares about?

Recommendations. There is a trade-off between sharing technologies *vs.* standards, within constraints that come from computational facilities. As standards are often constraining and/or hard to implement across facilities, one approach could be to focus on promoting standards of giving data from user to workflow systems/operators, i.e. a standard format for describing workflow input objects. Another recommendation is to develop a “marketplace” for workflow standards that would map them to parts of the workflow stack. The community would then be able to identify relevant standards for their systems and infrastructures, as well as identify gaps in current standards. Finally, there is a pressing need for defining a common vocabulary for components of the workflow stack and even the relationships between them.

4 AI Workflows

Artificial intelligence (AI) and machine learning (ML) methods are now mainstream in modern science. As computational power increases, including the recent achievement of exascale computing, AI/ML methods can at present, for example, generate highly accurate models that accelerate the rate of scientific discovery of complex problems running at very large scales. As a result, workflows increasingly integrate ML models to guide analysis, couple simulation and data analysis codes, and exploit specialized computing hardware (e.g., GPUs and neuromorphic chips) [34, 35]. These workflows inherently couple various types of tasks such as short ML inference, multi-node simulations, and long-running ML model training [36]. They are also often iterative and dynamic, with learning systems deciding in real time how to modify the workflow, e.g., by adding new simulations or changing the workflow all together. In this breakout session, participants focused on understanding the different characteristics of AI workflows, specifically the role of AI to create workflows or when used within workflows.

4.1 AI Workflows Characteristics

In the previous roadmap [6], discussions focused on identifying challenges inherent to workflows in which (most of) their tasks represent AI methods. The main challenges include the fine-grained data management and versioning features, heterogeneity of resources, integration to widely used ML framework, the iterative nature of ML processes, and the support of dynamic branching (Table 2). Although these challenges are still

relevant to the current landscape of AI workflows, the community experience with this ever-increasing class of workflows has identified novel characteristics that hinder the efficient execution of these workflow applications at large scales. For instance, managing many small files (e.g., image training sets) may significantly impair the performance of the shared filesystem. Although solutions such as NVMe (nonvolatile memory express) have significantly improved I/O throughput, the data management aspect in the software component still lacks optimized solutions to tackle the large volumes of data produced during workflow execution. Additionally, the volume of metadata generated for provenance may also become intractable.

In addition to the data management issue above, pseudo-random access to datasets leads to another major challenge in which optimizing access to microservices and performing data caching operations becomes a fundamental scheduling and/or resource provisioning problem. This challenge is aggravated by the unreliability of models, i.e., training processes often have a *human-in-the-loop* decision process that may result in a dramatic change to the shape of the task graph as the workflow execution evolves. Furthermore, coupling AI-coupled HPC workflows introduces additional challenges arising from the coupling of AI/ML models to traditional HPC workflows.

Recommendations. There is a need for a better understanding of the requirements of AI workflows. Although there exists a diverse set of AI workflow applications that could be leveraged as benchmarks or proxy/mini-apps, the complex specialized deployment (and requirements) of these workflows may prevent their large adoption that would enable comparative studies across computing facilities. Recently, the community has developed workflow benchmarks [37–39] that help to understand the requirements of traditional workflow applications; however, these benchmarks do not capture most of the characteristics intrinsic to AI workflows. Thus, there is a pressing need for creating a benchmark suite for representative AI workflows.

4.2 AI Workflows Categories and Motifs

The increasing adoption of AI/ML in modern science has not only enabled the development of novel applications using these methods [40] but also to improvements of systems and problem solving. In the workflows domain, the terminology around ‘AI workflows’ has become overloaded; thus, it is necessary to clearly and consistently define the different (sub)categories that encompass the distinct and unique use of AI/ML methods in the different aspects of workflow applications and systems. One approach to define these workflows into (sub)categories considering the role of AI would be: (i) *Workflows for AI* – the workflow is to develop the AI; (ii) *AI-enabled-workflows*. The latter in turn encapsulates two different sub-classes, viz., the *AI-integrated workflows*, where the workflow is about problem-solving with AI; and *AI-enhanced-workflow systems*, where the workflow engine is empowered with intelligence using a form of AI.

Within each of the above categories there should be subdivisions that could, for example, consider the dynamic nature of the workflow task graph (conditions, loops) or its adaptive response to events (task prioritization or preemption due to dynamic sampling) that could lead to different design and communication requirements. For the first two categories, the workflows could be classified into applications composed of (i) *inner*, (ii) *outer*, or (iii) *coupled* loops. Examples of inner loops include ML replacement of subroutines, adapting, and parallel training. Outer loops are characterized by reinforcement learning (RL) – there is a central “controller” that defines how the different steps (e.g., rollout and training) interact; and active learning – similar to RL in the sense that there are training, inference, and simulation components with a different class of sampling. Coupled loops can be represented by digital twins, which are a component of the workflow controlled by an AI method.

A recent characterization used common motifs of AI-HPC workflows [41] that considers workflows using AI and workflows. These motifs include: (i) AI models steering ensembles of simulations (e.g., advanced sampling, swarm methods); (ii) multistage (and typically multiscale and multiphysics) pipelining (e.g., molecule selection, virtual screening); (iii) inverse design including from observations, or determine causal factors (e.g., molecule or material design given properties such as structure to sequence); (iv) concurrent duality (e.g., concurrent HPC simulation and AI based digital twins); (v) distributed models and dynamic data (e.g., distributed AI based reduction/analysis coupled to HPC simulation, diverse models on edge-to-exascale infrastructure); and (vi) adaptive execution for training (e.g., HPO, NAS, LLM).

Recommendations. For each of the above (sub)categories, it is necessary to refine the motifs that would capture the unique requirements of each class; and develop benchmarks based on them that would then help catalyze the development of tailored solutions or example problems. An immediate activity includes spearheading a community effort to formulate a technical/white paper that would provide common terminology and definitions of categories of AI workflows and their unique requirements. As a long-term recommendation, the community needs to determine ways to smooth the integration (along with the deployment) paths across these categories and their associated components.

5 High Performance Data Management and In Situ Workflows

In situ workflows aim to overcome I/O limitations in HPC by sharing data between simulation, analysis, visualization and orchestration tasks as data are produced. The term “in situ” has become an umbrella covering approaches well beyond the seminal idea to distribute the load of the traditional post hoc analysis associated with a numerical simulation throughout the execution of that simulation [42]. Nowadays, in situ workflows facilitate data reduction, annotation, and transformation in different stages (e.g., data acquisition, simulation, analysis, visualization). These workflows can rely on multiple components, execution environments, and data transport methods to bypass the file system while delivering data between components. Additionally, modern in situ workflows exhibit different data production and consumption patterns (i.e., volume, frequency, structure) that need to co-exist in a coordinated and efficient manner. This leads to a need for high-performance management of the input, output, intermediate data, and metadata produced and consumed during the workflow execution. In this breakout session, participants focused on identifying challenges related to data management for in situ workflows, including how data abstractions may provide fine data management detached from the computing platform, and challenges related to the broader range of in situ workflows today (e.g., streaming and event-based workflows).

In situ workflows have evolved from the traditional simulation and analysis approach. In large-scale science, heavy computations are typically offloaded from the edge where data are produced (e.g., from a large scale scientific instrument such as a particle detector) to HPC infrastructures and cloud resources during burst conditions, leading to complex geographically distributed platforms with diverse performance characteristics. A major challenge in this scenario consists in transferring all this generated data from the edge to the processing facility. An approach to lower the pressure put on network resources is to employ in situ data reduction, an optimization technique in which the size of data is reduced without compromising the quality of the information it carries. This implies that this approach has to control and bound the loss of precision to ensure that end users will still have trust in the data. Approaches leveraging self-descriptive data and metadata could thus be used to define the relevance of pieces of data thanks to a tolerable error bound [43–45] Adaptive compression by regions of interest can also be used to reduce the size of data,

using techniques similar to those underlying Adaptive Mesh Refinement: regions where information is more important are compressed less than those where information has little scientific significance.

The need for high performance data management has significantly increased as modern science applications leverage heterogeneous resources for scaling their computations. The emergence of AI/ML workflows is intertwined with the imminent need for efficiency in situ data management, as these workflows increasingly produce and consume large volumes of data and have new motifs that profoundly differ from traditional workflows (Section 4). Differences in data formats, acquisition and pre-processing methods, training approaches, and provenance needs further exacerbate the complexity of enabling ML/AI for in situ analysis. Additionally, a key element in modern in situ workflows is the need to integrate them into the edge-to-cloud continuum. In this environment, data exchanges are typically performed through data objects in which the storage stack and data location are hidden from the application. Consequently, individual tasks create and work with objects, not files, and are not concerned with where those objects live. This enables storage-centric optimizations transparent to the end user to improve application performance. A potential solution to address the challenges and requirements of geographically distributed in situ workflows is to develop a data management layer abstraction for providing transparent data operations optimization and intelligent decisions for the workflow application. Ideally, workflow tasks would interact with this data management layer responsible for tracking the location of all these objects and performing intelligent data movements to place the necessary objects physically close to the computations using them. However, providing a single abstraction layer can be challenging when dealing with the requirements of a diverse community (e.g., different types of data, performance requisites, latency, consumption rates, availability needs, etc.). Thus, it is necessary to separate the data management paradigm from the workflow orchestration paradigm and move towards a *data-centric event-driven data plane for workflows*. Note that it is also necessary to detach data from its structure (e.g., file) and associated implementation (e.g., HDF5). Frameworks such as ADIOS2 [46] and Maestro [47], combined with self-descriptive data (metadata) capabilities and the publish/subscribe model, can be used to expose the data interaction mechanisms available to applications when building workflows.

Emerging in situ workflows involve streaming and event-based elements. Event-driven workflows have a strong need for tightly-coupled and high performing components that integrate HPC and streaming (especially for urgent computing) [48]. In these workflows, in situ computations are initiated based on events (e.g., changes to a variable or data, or access in a data repository). This class of workflows has been largely adapted to Cloud computing environments based on a service-oriented approach that uses streams of data for composing the workflows. In the HPC environment, streaming workflows are typically limited by policies that restrict connectivity to external data sources or long-running services. One approach to address these limitations is to couple a container-based platform (e.g., NERSC's Spin, OLCF's Slate) to HPC systems, in which services running on these platforms would act as a bridge between the outside world and the HPC ecosystem. Another limitation for enabling urgent computing on HPC is the need for specialized queues that can swiftly allocate resources for these types of jobs.

Recommendations. There is a need for determining what are the requirements for upcoming workflow applications, especially those integrating data or moving data across facilities. Dataflow and I/O contention benchmarks are needed to assist in the quantification of performance for workflow and data management solutions in upcoming scenarios. Current efforts [37, 49] mostly target traditional workflow motifs, and must be expanded to cover new applications (e.g., AI/ML workflows) and infrastructures (e.g., edge-to-cloud workflows). There is also a need for defining community-driven schemas to describe data and metadata, and to further provide an integrated view of the different views of the data associated to a workflow execution.

There is also a need to define a service-oriented composability approach for enabling urgent computing

workflows in the HPC ecosystem. To explore this, the community should conduct a comprehensive study to understand the specific needs for this class of applications for HPC, derive lessons learned from the service-oriented Cloud computing solutions, and engage with HPC computing facility operators to design a potential solution that would satisfy both the application's performance requirements as well as guarantee that facilities' security policies are still enforced.

6 HPC and Quantum Workflows

Most current workflow tools can operate over classical HPC, providing automated orchestration of tasks and data management. With the emergence of quantum computing (QC) there is a need to bridge these two computing models to improve the efficiency and potential impact of applications that could leverage the capabilities of each model. The need for a hybrid QC-HPC/Cloud approach is motivated for modern applications. In the quantum chemistry domain, for instance, researchers seek to understand and harness the quantum properties of atoms and systems around us. QC has also been increasingly used for ML processing and optimization – conversely, ML has been leveraged for the calibration of quantum processors.

In the current state of quantum computing, a.k.a. noisy intermediate-scale quantum (NISQ) era, there is a limited number of noisy physical qubits, thus it is paramount that quantum algorithms be efficient. Recently, variational quantum algorithms (VQAs) [50] have shown modest success in ML and optimization tasks. The approach requires a constant exchange of data between classical and quantum devices, which leads to a model in which quantum devices act as accelerators for classical computing. As a result, managing these sets of distributed, heterogeneous devices motivates the need for workflow solutions that can provide a common abstraction layer to interface with the multitude of specialized APIs and components provided by each platform.

Due to the novelty of the topic, this breakout session was structured with a series of short talks that underlined different aspects of quantum computing in workflows research, and identified a list of challenges associated to each of these aspects. Table 4 shows the list of topics. All presentation videos can be found at the WCI YouTube channel (https://www.youtube.com/playlist?list=PLAtmuqHEXrV0-l-mACm1K_4mfmG6yLnpN).

Topic	Presenter
Tensor Networks for Quantum Simulation	Artur Garcia Saez (BSC)
Advances in Hybrid Quantum-Classical High-Performance Computing	Stefan Andersson (ParTec AG), Mathias Pütz (ParTec AG)
Workflow Scheduling Using Quantum Devices	Justyna Zawalska (CYFRONET AGH)
Mechanisms for Enhancing Reliability/Recovery and Performance of Future Workloads	Ryan E. Grant (Queen's University)
HPCQC System Workflows	Martin Ruefenacht (LRZ)
Multi-core Quantum Computing	Aharon Brodutch (Entangled Networks)

Table 4: HPC and quantum workflows breakout topics.

6.1 State-of-the-art and Challenges

Recent efforts target the definition of languages for modeling computations between classical and quantum computing [51]. The open-source Covalent framework [16] provides mechanisms to manage experiments (expressed as workflows) and facilitate access to quantum devices, bridging them to HPC and cloud platforms. RosneT is a library for distributed, out-of-core block tensor algebra [52] built on top of the PyCOMPSs [53] programming model to transform tensor operations into a collection of tasks to be executed on HPC resources. QC has also been leveraged to solve well-known scheduling problems for traditional workflow applications [54].

Despite the above efforts, there are still several challenges that need to be addressed. For instance, (i) the reproducibility aspect in quantum computing requires domain expertise to understand the circuits and qubit topologies; (ii) the scheduling timescales significantly differ between classical and quantum computing – the latter is performed in the order of microseconds; (iii) there is a limited number of available resources, thus waiting time in queues can attain hours and costs to perform substantial computations may become excessive; (iv) programming each quantum device requires expert knowledge for each vendor API; (v) there are no standard representations for intermediate representations (IRs) for quantum programs, which are described as circuits; (vi) although some approaches bridge quantum devices to HPC and cloud systems, there is no tight integration between them (e.g., as between GPUs and CPUs).

Recommendations. The community is actively working in multiple objectives to make progress in the different topics of research. The foremost research directions include (i) the optimization of classical simulators based on tensor networks in classical HPC systems; (ii) the need to develop quantum systems with a component-based architecture to facilitate the integration with HPC and cloud environments; (iii) moving the hardware-aware software from the Quantum to the HPC system; (iv) reaching a consensus between QC as accelerators for classical computing (e.g., QPUs as accelerators to combine traditional HPC-QC workflows) or *vice versa*; (v) predicting optimizability of workflow decisions (e.g., how long will it take to optimize a QC workflow? Should a circuit be optimized?).

7 FAIR Computational Workflows

The original FAIR principles [55] laid a foundation for sharing and publishing data assets, emphasizing machine accessibility in that data and all other assets should be: (i) Findable – user of persistent identifies, cataloguing and indexing of data; (ii) Interoperable – machine processable metadata using standards; (iii) Accessible – clear access protocols to access (meta)data; (iv) Reusable – metadata standards, machine accessible usage license and provenance. When considering FAIR computational workflows, both data and software aspects need to be considered. FAIR data principles can be applied to, for example, workflow descriptions and specifications (typically via a domain specific language or API) that can be associated to metadata with Digital Object Identifiers (DOIs) or objects such as test data and parameter files. Workflow software objects, on the other hand, bring additional challenges such as reproducibility, usability, quality, maturity, etc. The recently published FAIR for research software principles [56] can then be applied to both workflows and workflow management systems. In this breakout session, participants focused on the software aspect of workflows and discussed best practices for research software registries and repositories, as well as approaches to build FAIR into workflow systems.

7.1 FAIR Workflow Repositories and Registries

The current state of the art for finding and accessing workflows consists of community repositories (e.g., nf-core [57], snakemake workflow catalog [58]), community platforms (e.g., nextflow tower [59]), data repositories (e.g., Zenodo [60], Dataverse [61]), registries (e.g., WorkflowHub [14], Dockstore [62]), and metadata frameworks (e.g., CWL [20], Workflow RO-Crate [63], Bioschemas profiles [64]). Registries and repositories provide curation and best practices for recording workflows, while metadata frameworks capture workflows description and their associated metadata (e.g., via canonical descriptions).

The notion that workflows can encompass both data and software brings several challenges when building FAIR workflow repositories and registries. In addition to capturing the workflow descriptions and (reference to) their input data, it is also necessary to capture metadata, containers, execution information, configurations, etc. The current approach adopted by the community is to separate the workflow definition from its execution – e.g., WorkflowHub records the workflow description and refers to Workflow-RO-Crates objects for execution information; Workflow-Run-RO-Crates collect the actual execution provenance. Although this approach has demonstrated success for traditional DAG- and cloud-based workflows, capturing workflow executions in the edge-to-HPC computing continuum is still an open question – mostly due to their intrinsic configurations, specialized architectures, and unique scientific instruments. The dynamic nature of emerging AI/ML workflows poses an additional challenge to capture executions that match the canonical description of the workflow.

An additional feature for registries would be to provide mechanisms to characterize workflows in terms of structures so that a user could explore similar solutions to their problem. Recent approaches on extracting metadata from software and research data repositories [30, 65, 66] or automated extraction of workflow patterns [67] could be leveraged to measure similarities across workflows. Past work [68] should be revisited. For instance, workflows could be compared by tags or labels, by the workflow structure (i.e., task graph), or by a stored relationship (e.g., sub-workflow, forked variant, etc.). There is also a need for defining persistent identifiers (PIDs) for workflows. DOIs are appropriate for workflows that are “completed and published”, i.e., as a snapshot of the workflow – similar to the publication process of research articles. There is a need for defining persistent identifiers (PIDs) for workflows that capture their complex nature.

Recommendations. Given the above, an imminent need is to define standards for capturing the metadata for workflows. This is critical for enabling automatic FAIR (unit/end-to-end) testing of workflows to support reuse and composability. There is also a need to expand the first set of recommendations for research software registries and repositories [69] for scientific workflows, especially attempting to address the need for workflow PIDs. A long-term activity is to consider new paradigms for workflow descriptions that could capture the emerging class of dynamic workflows.

7.2 Building FAIR into Workflow Management Systems

In addition to providing repositories and registries to store workflows and their associated metadata/information, it is crucial to empower workflow systems with capabilities to enable support to FAIR data and software throughout the workflow execution. To attain this objective, the community argues that two key features are needed: *standards* and *metadata*. Specifically, there is a need for a standard for expressing the inputs of the workflow and how to set them. Unfortunately, this issue is not novel and has not received much attention from the community. In fact, this issue is more a people problem than a technical problem, as is

frequently also the case when attempts to implement FAIR practices fall short [70]. Ideally, in terms of FAIRness, documentation is the most important asset for a workflow. It is very difficult to maintain software, but documentation lives on. On the description of the workflow itself, there are system agnostic languages (e.g., CWL and OpenWDL [71]) that have been adopted by several workflow systems to foster workflow portability, though these may be limited in the workflows that they can describe. The goal is to provide a mechanism to describe workflows so that they can be compared. Portability at the system level remains an open question, especially in edge and HPC environments.

The availability of metadata is another key feature necessary to enable FAIR within workflow systems. To this end, FAIR registration needs to be automatic and cannot introduce overhead in recording provenance information (i.e., the facts that link the inputs and outputs of the workflow) when running the workflow – recording of provenance information needs to be transparent and scalable [72]. Thus, FAIR data and FAIR workflows are intertwined. A solution would be to leverage Workflow-Run RO-Crate profiles, where the workflow system would export these profiles that would capture the provenance of an execution of a computational workflow. As a result, a coupling between the workflow and its associated data and metadata would be properly documented.

Recommendations. An immediate action is to continue the efforts to define methods for fostering portability across workflow systems, instead of defining the “standard” language for expressing workflows. Another key direction is to invest into people and communities rather than specific tools. The community strongly believes that the development of people’s and communities’ skills will outlast workflow systems, thus the incentive for FAIRness will be built intrinsically into these systems. Last, there is a need to assess the limitation of FAIR for different kinds of workflow types such as streaming and IoT workflows, or HPC workflows that are monolithic and tied to a particular architecture.

8 Workflows for Continuum and Cross-Facility Computing

Continuum and cross-facilities workflows are becoming more prevalent in the computational sciences. As these workflow paradigms have recently emerged, their definitions are evolving. The current understanding is that continuum workflows represent analysis pipelines that require continuous access to computing (e.g., urgent computing). Examples of workflows that fall under this category include in situ workflows, components that talk to each other within the same system (Section 5); geographical computing, i.e., edge-to-cloud-to-HPC computing; and high throughput computing, e.g., long-running computing campaigns over long periods of time. Cross-facility workflows represent analysis pipelines that hit more than a single site, which may include an experiment and a computing site, multiple experiment sites, or multiple computing facilities [8] – a site can be defined as a local compute, HPC, cloud, edge, campus cluster, and sensors at the edge. Cross-facility workflows can be seen as a solution for the needs of a continuum workflow (e.g., commercial cloud providers can be considered effectively cross-facility) offering resiliency for the computing needs of real time workflows. Another view is to consider continuum and cross-facility computing by the data plane, i.e., workflow systems should be able to tolerate different representations and underlying storage systems. In this breakout session, participants focused on identifying challenges to enable these categories of workflows and potential actions for the community.

8.1 Characteristics and Challenges

Data management is a major challenge in modern science, which involves operating data streams (e.g., from an instrument to a computing resource), data staging (e.g., processing fragments of data from different locations), or data movement (e.g., wide area transfers). The need for running workflows on multiple computing sites has increased as, for example, nowadays instruments can generate an ever-growing volume of data (in the order of TB/s) [73]. These workflows are characterized by their unique requirements to access and operate over computing resources regardless of the time or location sensitivity. For instance, existing computing facilities provide no mechanisms for scheduling and/or provisioning I/O capacity. Enabling a data-centric approach (i.e., doing computation on data quanta) may make the problem more tractable. To this end, it is necessary that tools to measure system contention/resource utilization and capture I/O and data movement performance are made available for measuring this end-to-end performance at multiple levels: computing system, across the network (local and wide area), and across the multiple compute and user facilities (e.g., node memory at one site to a file system at another site). Current systems perform matchmaking of task requirements (CPU/GPU/RAM) across multiple HPC platforms (e.g., RADICAL-Cybertools [74], HTCondor [75], etc.), however I/O requirements are neglected due to the lack of control over the I/O performance of an HPC machine at a given point in time.

A precondition to attain the envisaged level of fine-grained workflow orchestration described above is to define the requirements of a workflow task. Specifically, what metadata does the workflow task need to have to request resources (computing, storage, network) from the appropriate site? To answer this question, it is necessary to convey that tasks are less portable than commonly assumed (including containers). The process for labelling every single resource a task needs is still an open question, and requirements may significantly diverge considering the targeted computing environment (e.g., compiler optimization for a determined architecture, data location and network and I/O bandwidths, policies, etc.).

In addition to the above challenges, cross-facility workflows also face challenges associated to autonomous administration domains, i.e., different policies and security models at different sites (authorization, access control), different software stacks, etc. Approaches for providing resource information in a machine-readable format (e.g., NERSC's superfacility project [26]) or enabling federated identification (e.g., DOE's OneID [76]) are a first step for enabling distributed workflow orchestration and intelligent decisions at the workflow management system level. A collaboration between the Computational Science Initiative (CSI), the Center for Functional Nanomaterials (CFN), and the National Synchrotron Light Source (NSLS-II) attempt to facilitate using larger institutional compute resources to support the beamlines. The idea is to go from giving users access to beamlines and sending them home with their data, to giving users access to experiments and sending them home with answers to their science questions. Multi-site workflows are a mainstream component to this effort.

A different approach is letting users embrace and exploit heterogeneity from the design phase through *hybrid workflows* [77]. Traditional workflow models, which describe steps and data dependencies, can be flanked by a topology of execution locations, encoding execution environments, available resources, and communication channels. Users can then explicitly map the different steps of the workflow onto different locations of the topology in an N-to-M relationship. Hybrid workflows are heterogeneous by design and do not require a homogeneous representation of locations' capabilities. As a result, site-specific plugins can be used to handle orchestration aspects such as authentication, scheduling, data transfers, and task offloading. Hybrid workflows have already successfully orchestrated cloud-HPC and cross-HPC distributed workflows using batched [77] and interactive [78] execution paradigms. The main drawback is that the

additional complexity introduced in the design phase steepens the learning curve for domain experts with little knowledge of distributed execution environments.

Recommendations. Continuum and cross-facility workflows are becoming mainstream in modern science. These new classes of workflows present, in addition to the typical challenges inherent to computing on a single site, new challenges that require coordination and cooperation among computing and experimental facilities. An immediate action is to enable tracking the metadata associated with tasks in workflow across different sites. To this end, it is necessary to define a standard/specification for task descriptions, including a way to represent the overall I/O requirements of an entire workflow. There is also a need for a community effort to define what a computing site needs to provide to enable a workflow orchestrator to make intelligent decisions about where to place tasks in a workflow. Another recommendation is to factor in the need for cross-facility computing from the conceptualization/design phase of computer systems and experimental facilities – typically, personnel involved in the design of experiment facilities are not necessarily computing experts.

References

- [1] T. Ben-Nun, T. Gamblin, D. S. Hollman, H. Krishnan, and C. J. Newburn, “Workflows are the new applications: Challenges in performance, portability, and productivity,” in *2020 IEEE/ACM International Workshop on Performance, Portability and Productivity in HPC (P3HPC)*. IEEE, 2020, pp. 57–69.
- [2] R. M. Badia Sala, E. Ayguadé Parra, and J. J. Labarta Mancho, “Workflows for science: A challenge when facing the convergence of hpc and big data,” *Supercomputing frontiers and innovations*, vol. 4, no. 1, pp. 27–47, 2017.
- [3] R. Ferreira da Silva, R. Filgueira, I. Pietri, M. Jiang, R. Sakellariou, and E. Deelman, “A characterization of workflow management systems for extreme-scale applications,” *Future Generation Computer Systems*, vol. 75, pp. 228–238, 2017. doi: 10.1016/j.future.2017.02.026
- [4] I. Foster, “Coding the continuum,” in *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2019. doi: 10.1109/IPDPS.2019.00011
- [5] I. Altintas, I. Perez, D. Mishin, A. Trouillaud, C. Irving, J. Graham, M. Tatineni, T. DeFanti, S. Strande, L. Smarr *et al.*, “Towards a dynamic composability approach for using heterogeneous systems in remote sensing,” in *2022 IEEE 18th International Conference on e-Science (e-Science)*. IEEE, 2022, pp. 336–345.
- [6] R. Ferreira da Silva, H. Casanova, K. Chard, I. Altintas, R. M. Badia, B. Balis, T. Coleman, F. Coppens, F. Di Natale, B. Enders, T. Fahringer, R. Filgueira, G. Fursin, D. Garijo, C. Goble, D. Howell, S. Jha, D. S. Katz, D. Laney, U. Leser, M. Malawski, K. Mehta, L. Pottier, J. Ozik, J. L. Peterson, L. Ramakrishnan, S. Soiland-Reyes, D. Thain, and M. Wolf, “A community roadmap for scientific workflows research and development,” in *2021 IEEE Workshop on Workflows in Support of Large-Scale Science (WORKS)*, 2021. doi: 10.1109/WORKS54523.2021.00016 pp. 81–90.
- [7] F. Poeschel, J. E. W. F. Godoy, N. Podhorszki, S. Klasky, G. Eisenhauer, P. E. Davis, L. Wan, A. Gainaru, J. Gu *et al.*, “Transitioning from file-based hpc workflows to streaming data pipelines with openpmd and adios2,” *Driving Scientific and Engineering Discoveries Through the Integration of Experiment, Big Data, and Modeling and Simulation: 21st Smoky Mountains Computational Sciences and Engineering, SMC 2021, Virtual Event, October 18-20, 2021, Revised Selected Papers*, pp. 99–118, 2022.
- [8] K. B. Antypas, D. Bard, J. P. Blaschke, R. S. Canon, B. Enders, M. A. Shankar, S. Somnath, D. Stansberry, T. D. Uram, and S. R. Wilkinson, “Enabling discovery data science through cross-facility workflows,” in *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 2021, pp. 3671–3680.
- [9] S. Cohen-Boulakia, K. Belhajjame, O. Collin, J. Chopard, C. Froidevaux, A. Gaignard, K. Hinsén, P. Larmande, Y. Le Bras, F. Lemoine *et al.*, “Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities,” *Future Generation Computer Systems*, vol. 75, pp. 284–298, 2017.
- [10] A. Al-Saadi, D. H. Ahn, Y. Babuji, K. Chard, J. Corbett, M. Hategan, S. Herbein, S. Jha, D. Laney, A. Merzky *et al.*, “Exaworks: Workflows for exascale,” in *2021 IEEE Workshop on Workflows in Support of Large-Scale Science (WORKS)*. IEEE, 2021, pp. 50–57.

- [11] C. Goble, S. Cohen-Boulakia, S. Soiland-Reyes, D. Garijo, Y. Gil, M. R. Crusoe, K. Peters, and D. Schober, “Fair computational workflows,” *Data Intelligence*, vol. 2, no. 1-2, pp. 108–121, 2020.
- [12] “Workflows Community Summit 2022,” <https://workflows.community/summits/2022>, 2022.
- [13] “Workflows Community Initiative,” <https://workflows.community>, 2023.
- [14] C. Goble, S. Soiland-Reyes, F. Bacall, S. Owen, A. Williams, I. Eguinoa, B. Driesbeke, S. Leo, L. Pireddu, L. Rodríguez-Navas *et al.*, “Implementing fair digital objects in the eosco-life workflow collaboratory,” *Zenodo*, 2021.
- [15] “eFlows4HPC: Enabling Dynamic and Intelligent Workflows in the Future EuroHPC Ecosystem,” <https://eflows4hpc.eu>, 2023.
- [16] “Covalent: Open Source Workflow Orchestration for Heterogenous Computing,” <https://www.covalent.xyz>, 2023.
- [17] R. Ferreira da Silva, H. Casanova, K. Chard, D. Laney, D. Ahn, S. Jha, C. Goble, L. Ramakrishnan, L. Peterson, B. Enders, D. Thain, I. Altintas, Y. Babuji, R. Badia, V. Bonazzi, T. Coleman, M. Crusoe, E. Deelman, F. Di Natale, P. Di Tommaso, T. Fahringer, R. Filgueira, G. Fursin, A. Ganose, B. Gruning, D. S. Katz, O. Kuchar, A. Kupresanin, B. Ludascher, K. Maheshwari, M. Mattoso, K. Mehta, T. Munson, J. Ozik, T. Peterka, L. Pottier, T. Randles, S. Soiland-Reyes, B. Tovar, M. Turilli, T. Uram, K. Vahi, M. Wilde, M. Wolf, and J. Wozniak, “Workflows Community Summit: Bringing the Scientific Workflows Community Together,” Mar. 2021. doi: 10.5281/zenodo.4606958
- [18] R. Ferreira da Silva, H. Casanova, K. Chard, T. a. Coleman, D. Laney, D. Ahn, S. Jha, D. Howell, S. Soiland-Reys, I. Altintas, D. Thain, R. Filgueira, Y. Babuji, R. M. Badia, B. Balis, S. Caino-Lores, S. Callaghan, F. Coppens, M. R. Crusoe, K. De, F. Di Natale, T. M. A. Do, B. Enders, T. Fahringer, A. Fouilloux, G. Fursin, A. Gaignard, A. Ganose, D. Garijo, S. Gesing, C. Goble, A. Hasan, S. Huber, D. S. Katz, U. Leser, D. Lowe, B. Ludaescher, K. Maheshwari, M. Malawski, R. Mayani, K. Mehta, A. Merzky, T. Munson, J. Ozik, L. Pottier, S. Ristov, M. Roozmeh, R. Souza, F. Suter, B. Tovar, M. Turilli, K. Vahi, A. Vidal-Torreira, W. Whitcup, M. Wilde, A. Williams, M. Wolf, and J. Wozniak, “Workflows Community Summit: Advancing the State-of-the-art of Scientific Workflows Management Systems Research and Development,” Jun. 2021. doi: 10.5281/zenodo.4915801
- [19] R. Ferreira da Silva, K. Chard, H. Casanova, D. Laney, D. Ahn, S. Jha, W. E. Allcock, G. Bauer, D. Duplyakin, B. Enders, T. M. Heer, E. Lançon, S. Sanielevici, and K. Sayers, “Workflows Community Summit: Tightening the Integration between Computing Facilities and Scientific Workflows,” Jan. 2022. doi: 10.5281/zenodo.5815332
- [20] M. R. Crusoe, S. Abeln, A. Iosup, P. Amstutz, J. Chilton, N. Tijanić, H. Ménager, S. Soiland-Reyes, B. Gavrilović, C. Goble *et al.*, “Methods included: Standardizing computational reuse and portability with the common workflow language,” *Communications of the ACM*, vol. 65, no. 6, pp. 54–63, 2022.
- [21] A. Gallo, I. Claydon, E. Tucker, and R. Arthur, “Industrial experience deploying heterogeneous platforms for use in multi-modal power systems design workflows,” in *Accelerating Science and Engineering Discoveries Through Integrated Research Infrastructure for Experiment, Big Data, Modeling and*

- Simulation: 22nd Smoky Mountains Computational Sciences and Engineering Conference, SMC 2022, Virtual Event, August 23–25, 2022, Revised Selected Papers.* Springer, 2023, pp. 257–273.
- [22] D. Garijo, P. Alper, K. Belhajjame, O. Corcho, Y. Gil, and C. Goble, “Common motifs in scientific workflows: An empirical analysis,” *Future generation computer systems*, vol. 36, pp. 338–351, 2014.
- [23] “Global Alliance for Genomics & Health,” <https://www.ga4gh.org/>, 2023.
- [24] “GA4GH Cloud Work Stream,” <https://ga4gh-cloud.github.io/>, 2023.
- [25] “Pulsar-Network,” <https://pulsar-network.readthedocs.io/en/latest/index.html>, 2023.
- [26] B. Enders, D. Bard, C. Snavely, L. Gerhardt, J. Lee, B. Totzke, K. Antypas, S. Byna, R. Cheema, S. Cholia *et al.*, “Cross-facility science with the superfacility project at lbl,” in *2020 IEEE/ACM 2nd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, 2020, pp. 1–7.
- [27] F. Lehmann, J. Bader, F. Tschirpke, L. Thamsen, and U. Leser, “How workflow engines should talk to resource managers: A proposal for a common workflow scheduling interface,” in *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, Bangalore, India, 2023.
- [28] M. Hilbrich, S. Müller, S. Kulagina, C. Lazik, N. De Mecquenem, and L. Grunske, “A consolidated view on specification languages for data analysis workflows,” in *Leveraging Applications of Formal Methods, Verification and Validation. Software Engineering: 11th International Symposium, ISoLA 2022, Rhodes, Greece, October 22–30, 2022, Proceedings, Part II*. Springer, 2022, pp. 201–215.
- [29] H. Suetake, T. Tanjo, M. Ishii, B. P. Kinoshita, T. Fujino, T. Hachiya, Y. Kodama, T. Fujisawa, O. Ogasawara, A. Shimizu *et al.*, “Sapporo: a workflow execution service that encourages the reuse of workflows in various languages in bioinformatics,” *F1000Research*, vol. 11, no. 889, p. 889, 2022.
- [30] T. J. Skluzacek, R. Wong, Z. Li, R. Chard, K. Chard, and I. Foster, “A serverless framework for distributed bulk metadata extraction,” in *Proceedings of the 30th International Symposium on High-Performance Parallel and Distributed Computing*, 2021, pp. 7–18.
- [31] S. Blanas and S. Byna, “Towards exascale scientific metadata management,” *arXiv preprint arXiv:1503.08482*, 2015.
- [32] “Workflow Run RO-Crate,” <https://www.researchobject.org/workflow-run-crate/>, 2023.
- [33] F. Z. Khan, S. Soiland-Reyes, R. O. Sinnott, A. Lonie, C. Goble, and M. R. Crusoe, “Sharing interoperable workflow provenance: A review of best practices and their practical application in cwlprov,” *GigaScience*, vol. 8, no. 11, p. giz095, 2019.
- [34] A. Brace, H. Lee, H. Ma, A. Trifan, M. Turilli, I. Yakushin, T. Munson, I. Foster, S. Jha, and A. Ramanathan, “Achieving 100x faster simulations of complex biological phenomena by coupling ml to hpc ensembles,” *arXiv preprint arXiv:2104.04797*, 2021.

- [35] J. Ejarque, R. M. Badia, L. Albertin, G. Aloisio, E. Baglione, Y. Becerra, S. Boschert, J. R. Berlin, A. D’Anca, D. Elia, F. Exertier, S. Fiore, J. Flich, A. Folch, S. J. Gibbons, N. Koldunov, F. Lordan, S. Lorito, F. Løvholt, J. Macías, F. Marozzo, A. Michelini, M. Monterrubio-Velasco, M. Pienkowska, J. de la Puente, A. Queralt, E. S. Quintana-Ortí, J. E. Rodríguez, F. Romano, R. Rossi, J. Rybicki, M. Kupczyk, J. Selva, D. Talia, R. Tonini, P. Trunfio, and M. Volpe, “Enabling dynamic and intelligent workflows for hpc, data analytics, and ai convergence,” *Future Generation Computer Systems*, vol. 134, pp. 414–429, 2022. doi: <https://doi.org/10.1016/j.future.2022.04.014>
- [36] S. Jha, V. R. Pascuzzi, and M. Turilli, “Ai-coupled hpc workflows,” *arXiv preprint arXiv:2208.11745*, 2022.
- [37] T. Coleman, H. Cansanova, K. Maheshwari, L. Pottier, S. R. Wilkinson, J. Wozniak, F. Suter, M. Shankar, and R. Ferreira da Silva, “WfBench: Automated Generation of Scientific Workflow Benchmarks,” in *2022 IEEE/ACM International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS)*, 2022. doi: 10.1109/PMBS56514.2022.00014 pp. 100–111.
- [38] E. Larssonneur, J. Mercier, N. Wiart, E. L. Floch, O. Delhomme, and V. Meyer, “Evaluating workflow management systems: A bioinformatics use case,” in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2018. doi: 10.1109/BIBM.2018.8621141 pp. 2773–2775.
- [39] “CWL Benchmarks from 4DN-DCIC,” <https://github.com/SooLee/Benchmark>, 2023.
- [40] B. J. Oakes, M. Famelis, and H. Sahraoui, “Building domain-specific machine learning workflows: A conceptual framework for the state-of-the-practice,” *arXiv preprint arXiv:2203.08638*, 2022.
- [41] W. Bethel, A. Gainaru, F. Suter, M. Emani, F. Wang, and S. Jha, “A survey of the landscape of AI-hpc workflows,” 2023.
- [42] H. Childs, S. D. Ahern, J. Ahrens, A. C. Bauer, J. Bennett, E. W. Bethel, P.-T. Bremer, E. Brugger, J. Cottam, M. Dorier *et al.*, “A terminology for in situ visualization and analysis systems,” *The International Journal of High Performance Computing Applications*, vol. 34, no. 6, pp. 676–691, 2020.
- [43] M. Ainsworth, O. Tugluk, B. Whitney, and S. Klasky, “Multilevel Techniques for Compression and Reduction of Scientific Data-Quantitative Control of Accuracy in Derived Quantities,” *SIAM Journal on Scientific Computing*, vol. 41, no. 4, pp. A2146–A2171, 2019. doi: 10.1137/18M1208885
- [44] T. Banerjee, J. Choi, J. Lee, Q. Gong, J. Chen, S. Klasky, A. Rangarajan, and S. Ranka, “Scalable Hybrid Learning Techniques for Scientific Data Compression,” 2022.
- [45] S. Di and F. Cappello, “Fast Error-Bounded Lossy HPC Data Compression with SZ,” in *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2016. doi: 10.1109/IPDPS.2016.11 pp. 730–739.
- [46] W. F. Godoy, N. Podhorszki, R. Wang, C. Atkins, G. Eisenhauer, J. Gu, P. Davis, J. Choi, K. Geraschewski, K. Huck, A. Huebl, M. Kim, J. Kress, T. Kurc, Q. Liu, J. Logan, K. Mehta, G. Ostrouchov, M. Parashar, F. Poeschel, D. Pugmire, E. Suchyta, K. Takahashi, N. Thompson, S. Tsutsumi, L. Wan, M. Wolf, K. Wu, and S. Klasky, “Adios 2: The adaptable input output system. a framework for high-performance data management,” *SoftwareX*, vol. 12, 2020. doi: 10.1016/j.softx.2020.100561

- [47] C. Haine, U.-U. Haus, M. Martinasso, D. Pleiter, F. Tessier, D. Sarmany, S. Smart, T. Quintino, and A. Tate, “A middleware supporting data movement in complex and software-defined storage and memory architectures,” in *High Performance Computing*. Springer International Publishing, 2021, pp. 346–357.
- [48] G. P. S. Gibb, N. Brown, R. W. Nash, M. Mendes, S. Monedero, H. D. Fidalgo, J. R. Cisneros, A. Cardil, and M. Kontak, “A bespoke workflow management system for data-driven urgent hpc,” in *2020 IEEE/ACM HPC for Urgent Decision Making (UrgentHPC)*, 2020. doi: 10.1109/UrgentHPC51945.2020.00007 pp. 10–20.
- [49] “WfBG: Workflow Benchmarking Group,” <https://workflows.community/groups/benchmarking/>, 2023.
- [50] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, “Variational quantum algorithms,” *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.
- [51] B. Weder, U. Breitenbücher, F. Leymann, and K. Wild, “Integrating quantum computing into workflow modeling and execution,” in *2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)*. IEEE, 2020, pp. 279–291.
- [52] S. Sánchez-Ramírez, J. Conejero, F. Lordan, A. Queralt, T. Cortes, R. M. Badia, and A. Garcia-Saez, “Rosnet: A block tensor algebra library for out-of-core quantum computing simulation,” in *2021 IEEE/ACM Second International Workshop on Quantum Computing Software (QCS)*. IEEE, 2021, pp. 1–8.
- [53] E. Tejedor, Y. Becerra, G. Alomar, A. Queralt, R. M. Badia, J. Torres, T. Cortes, and J. Labarta, “Py-compss: Parallel computational workflows in python,” *The International Journal of High Performance Computing Applications*, vol. 31, no. 1, pp. 66–82, 2017.
- [54] “Workflow scheduling using quantum devices,” https://github.com/qc-lab/workflows-dwave/blob/master/notebooks/Workflows_Demo.ipynb, 2022.
- [55] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, “The fair guiding principles for scientific data management and stewardship,” *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.
- [56] N. P. C. Hong, D. S. Katz, M. Barker, A.-L. Lamprecht, C. Martinez, F. E. Psomopoulos, J. Harrow, L. J. Castro, M. Gruenpeter, P. A. Martinez *et al.*, “Fair principles for research software (fair4rs principles),” *Research Data Alliance*, 2022. doi: 10.15497/RDA00068
- [57] P. A. Ewels, A. Peltzer, S. Fillinger, H. Patel, J. Alneberg, A. Wilm, M. U. Garcia, P. Di Tommaso, and S. Nahnsen, “The nf-core framework for community-curated bioinformatics pipelines,” *Nature biotechnology*, vol. 38, no. 3, pp. 276–278, 2020.
- [58] “Snakemake workflow catalog,” <https://snakemake.github.io/snakemake-workflow-catalog/>, 2023.
- [59] “Nextflow tower,” <https://tower.nf>, 2023.
- [60] “Zenodo,” <https://zenodo.org>, 2023.

- [61] A. Trisovic, P. Durbin, T. Schlatter, G. Durand, S. Barbosa, D. Brooke, and M. Crosas, “Advancing computational reproducibility in the dataverse data repository platform,” in *Proceedings of the 3rd International Workshop on Practical Reproducible Evaluation of Computer Systems*, 2020, pp. 15–20.
- [62] “Dockstore,” <https://dockstore.org>, 2023.
- [63] “Workflow RO-Crate,” <https://about.workflowhub.eu/Workflow-RO-Crate>, 2023.
- [64] “Bioschema: Computational workflow profile,” <https://bioschemas.org/profiles/ComputationalWorkflow/1.0-RELEASE>, 2021.
- [65] R. Filgueira and D. Garijo, “Inspect4py: a knowledge extraction framework for python code repositories,” in *Proceedings of the 19th International Conference on Mining Software Repositories*, 2022, pp. 232–236.
- [66] A. Kelley and D. Garijo, “A framework for creating knowledge graphs of scientific software metadata,” *Quantitative Science Studies*, vol. 2, no. 4, pp. 1423–1446, 2021.
- [67] T. Coleman, H. Casanova, and R. Ferreira da Silva, “Wfchef: Automated generation of accurate scientific workflow generators,” in *17th IEEE eScience Conference*, 2021. doi: 10.1109/eScience51609.2021.00026 pp. 159–168.
- [68] J. Starlinger, B. Brancotte, S. Cohen-Boulakia, and U. Leser, “Similarity search for scientific workflows,” *Proceedings of the VLDB Endowment (PVLDB)*, vol. 7, no. 12, pp. 1143–1154, 2014.
- [69] D. Garijo, H. Ménager, L. Hwang, A. Trisovic, M. Hucka, T. Morrell, and A. Allen, “Nine best practices for research software registries and repositories,” *PeerJ Computer Science*, vol. 8, p. e1023, 2022.
- [70] S. R. Wilkinson, G. Eisenhauer, A. J. Kapadia, K. Knight, J. Logan, P. Widener, and M. Wolf, “F*** workflows: when parts of FAIR are missing,” in *2022 IEEE 18th International Conference on e-Science (e-Science)*, 2022. doi: 10.1109/eScience55777.2022.00090 pp. 507–512.
- [71] “Openwdl,” <https://openwdl.org>, 2023.
- [72] R. Sirvent, J. Conejero, F. Lordan, J. Ejarque, L. Rodríguez-Navas, J. M. Fernández, S. Capella-Gutiérrez, and R. M. Badia, “Automatic, efficient and scalable provenance registration for fair hpc workflows,” in *2022 IEEE/ACM Workshop on Workflows in Support of Large-Scale Science (WORKS)*. IEEE, 2022, pp. 1–9.
- [73] J. E. McClure, J. Yin, R. T. Armstrong, K. C. Maheshwari, S. Wilkinson, L. Vlcek, Y. Da Wang, M. A. Berrill, and M. Rivers, “Toward Real-Time Analysis of Synchrotron Micro-Tomography Data: Accelerating Experimental Workflows with AI and HPC,” in *Driving Scientific and Engineering Discoveries Through the Convergence of HPC, Big Data and AI*, J. Nichols, B. Verastegui, A. B. Maccabe, O. Hernandez, S. Parete-Koon, and T. Ahearn, Eds. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-63393-6_15. ISBN 978-3-030-63393-6 pp. 226–239.
- [74] V. Balasubramanian, S. Jha, A. Merzky, and M. Turilli, “Radical-cybertools: Middleware building blocks for scalable science,” *arXiv preprint arXiv:1904.03085*, 2019.

- [75] D. Thain, T. Tannenbaum, and M. Livny, “Distributed computing in practice: the condor experience,” *Concurrency and computation: practice and experience*, vol. 17, no. 2-4, pp. 323–356, 2005.
- [76] “OneID,” <https://eams-hub.oneid.energy.gov/hub>, 2023.
- [77] I. Colonnelli, B. Cantalupo, I. Merelli, and M. Aldinucci, “StreamFlow: cross-breeding cloud with HPC,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1723–1737, 2021. doi: 10.1109/TETC.2020.3019202
- [78] I. Colonnelli, M. Aldinucci, B. Cantalupo, L. Padovani, S. Rabellino, C. Spampinato, R. Morelli, R. Di Carlo, N. Magini, and C. Cavazzoni, “Distributed workflows with Jupyter,” *Future Generation Computer Systems*, vol. 128, pp. 282–298, 2022. doi: 10.1016/j.future.2021.10.007

Appendix A: Participants and Contributors

First Name	Last Name	Affiliation
Ilkay	Altintas	UC San Diego
Stefan	Andersson	ParTec AG
William	Arndt	National Energy Research Scientific Computing Center
Juan	Aznar	University of Innsbruck
Jonathan	Bader	TU Berlin
Rosa M	Badia	Barcelona Supercomputing Center
Venkat	Bala	Agnostiq
Bartosz	Balis	AGH University of Science and Technology
Debbie	Bard	National Energy Research Scientific Computing Center
Chris	Blanton	NOAA-GFDL
Timo	Bremer	Lawrence Livermore National Laboratory
Aharon	Brodutch	Entangled Networks
Ben	Brown	DOE ASCR
Paul	Brunk	University of Georgia
Ian	Buckley	Agnostiq
Silvina	Caino-Lores	University of Tennessee
Henri	Casanova	University of Hawaii at Manoa
Alba	Cervera-Lierta	Barcelona Supercomputing Center
Kyle	Chard	University of Chicago
Justin	Chigu	Egypt Japan University of Science and Technology
Taina	Coleman	University of Southern California
Nick	Collier	Argonne National Laboratory
Iacopo	Colonnelli	Università degli Studi di Torino
Frederik	Coppens	ELIXIR
Michael	Crusoe	CWL Project / VU Amsterdam / ELIXIR NL & DE
Will	Cunningham	Agnostiq
Bruno	de Paula Kinoshita	Barcelona Supercomputing Center
Paolo	Di Tommaso	Seqera Labs
Charles	Doutriaux	Lawrence Livermore National Lab
Matthew	Downton	NCI Australia
Wael	Elwasif	Oak Ridge National Laboratory
Bjoern	Enders	NERSC
Chris	Erdmann	Michael J. Fox Foundation
Thomas	Fahringer	University of Innsbruck
Rafael	Ferreira da Silva	Oak Ridge National Laboratory
Ludmilla	Figueiredo	German Centre for Integrative Biodiversity Research
Rosa	Filgueira	University of St. Andrews
Martin	Foltin	HP Labs
Anne	Fouilloux	Simula Research Laboratory
Luiz	Gadelha	University of Jena
Andy	Gallo	GE Research
Artur	Garcia	Barcelona Supercomputing Center
Daniel	Garijo	Universidad Politécnica de Madrid
Roman	Gerlach	Friedrich-Schiller-University Jena, Germany
Carole	Goble	The University of Manchester
Ryan	Grant	Queen's University
Samuel	Grayson	University of Illinois at Urbana-Champaign
Patricia	Grubel	Los Alamos National Laboratory
Johan	Gustafsson	Australian BioCommons, University of Melbourne

Valerie	Hayot	University of Chicago
Oscar	Hernandez	ORNL
Marcus	Hilbrich	Humboldt-Universität zu Berlin
Shantenu	Jha	Brookhaven National Laboratory / Rutgers University
Annmary	Justine	Hewlett Packard Enterprise
Daniel S.	Katz	University of Illinois at Urbana-Champaign
Ian	Laflotte	SAIC / NOAA/ GFDL
Daniel	Laney	Lawrence Livermore National Laboratory
Fabian	Lehmann	Humboldt-Universität zu Berlin
Andre	Luckow	Ludwig-Maximilian University Munich
Jakob	Luettgau	University of Tennessee
Ketan	Maheshwari	Oak Ridge National Laboratory
Motohiko	Matsuda	RIKEN
Doriana	Medic	University of Turin
Pete	Mendygral	HPE
Marek	Michalewicz	Sano Centre for Personalized Computational Medicine
Jorji	Nonaka	RIKEN R-CCS
Manish	Parashar	University of Utah
Maciej	Pawlik	University of Science and Technology AGH
Loïc	Pottier	Lawrence Livermore National Laboratory
Line	Pouchard	Brookhaven National Lab
Mathias	Pütz	ParTec AG
Santosh Kumar	Radha	Agnostiq Inc.
Lavanya	Ramakrishnan	LBNL
Sashko	Ristov	University of Innsbruck
Paul	Romano	Argonne National Laboratory
Kelly	Rosa Braghetto	University of São Paulo
Daniel	Rosendo	Inria
Martin	Ruefenacht	Leibniz Supercomputing Centre
Katarzyna	Rycerz	II & Cyfronet AGH
Nishant	Saurabh	Utrecht University
Volodymyr	Savchenko	EPFL
Martin	Schulz	TU-Munich and LRZ
Christine	Simpson	ALCF
Raül	Sirvent	Barcelona Supercomputing Center
Tyler	Skluzacek	Oak Ridge National Laboratory
Stian	Soiland-Reyes	The University of Manchester
Renan	Souza	Oak Ridge National Lab
Sreenivas Rangan	Sukumar	HPE
Ziheng	Sun	George Mason University
Alan	Sussman	University of Maryland
Frederic	Suter	ORNL
Douglas	Thain	Notre Dame University
Mikhail	Titov	Brookhaven National Laboratory
Benjamin	Tovar	Univeristy of Notre Dame
Aalap	Tripathy	Hewlett Packard Enterprise
Matteo	Turilli	BNL; Rutgers University
Bartosz	Tużnik	ICM, University of Warsaw
Nick	Tyler	NERSC
Thomas	Uram	Argonne National Laboratory
Hubertus	van Dam	Brookhaven National Laboratory
Aurelio	Vivas	Universidad de los Andes
Logan	Ward	Argonne National Laboratory
Patrick	Widener	Oak Ridge National Laboratory

Sean	Wilkinson	Oak Ridge National Laboratory
Justyna	Zawalska	AGH University of Science and Technology, ACC Cyfronet
Mahnoor	Zulfiqar	Friedrich Schiller University

Appendix B: Agenda

Day 1 – November 29, 2022

Time	Topic
11:00-11:05am EST	Welcome and Introductions <i>Rafael Ferreira da Silva (Oak Ridge National Laboratory)</i>
11:05-11:10am EST	The Workflows Community Initiative overview
11:10-11:20am EST	Remembering the 2021 Roadmap <i>Rafael Ferreira da Silva (Oak Ridge National Laboratory)</i>
11:20-11:50am EST	Lightning talks • Topic 1: Specifications, standards, and APIs <i>Daniel Laney (LLNL), Kyle Chard (UChicago)</i> • Topic 2: AI workflows <i>Shantenu Jha (BNL), Timo Bremer (LLNL)</i> • Topic 3: High performance data management and in situ workflows <i>Silvina Caino-Lores (UTK), Frédéric Suter (ORNL)</i>
11:50am-noon EST	<i>10min Break</i>
noon-1:30pm EST	Breakout sessions • Topic 1: Specifications, standards, and APIs • Topic 2: AI workflows • Topic 3: High performance data management and in situ workflows
1:30-1:45pm EST	<i>15min Break</i>
1:45-2:00pm EST	Reports from breakout sessions

Day 2 – November 30, 2022

Time	Topic
11:00-11:10am EST	Welcome and Introductions <i>Rafael Ferreira da Silva (Oak Ridge National Laboratory)</i>
11:10-11:45am EST	Lightning talks • Topic 4: HPC and Quantum workflows <i>Rosa Badia (BSC), Ian Buckley (Agnostiq), Venkat Bala (Agnostiq)</i> • Topic 5: FAIR workflows <i>Carole Goble (UManchester), Daniel S. Katz (UIUC)</i> • Topic 6: Workflows for continuum and cross-facility computing <i>Debbie Bard (NERSC), Tom Uram (ANL), Nick Tyler (NERSC)</i>
11:45am-noon EST	<i>15min Break</i>
noon-12:15pm EST	Invited talk: A Translational Perspective on End-to-End Workflows <i>Manish Parashar (Utah)</i>

12:15-1:35pm EST

Breakout sessions

- Topic 4: HPC and Quantum workflows
 - Tensor Networks for Quantum Simulation
Artur Garcia Saez (BSC)
 - Advances in Hybrid Quantum-Classical High-Performance Computing
Stefan Andersson (ParTec AG), Mathias Pütz (ParTec AG)
 - Workflow Scheduling Using Quantum Devices
Justyna Zawalska (Academic Computer Centre CYFRONET AGH)
 - Mechanisms for Enhancing Reliability/Recovery and Performance of Future Workloads
Ryan E. Grant (Queen's University)
 - HPCQC System Workflows
Martin Ruefenacht (Leibniz Supercomputing Centre)
 - Multi-Core Quantum Computing
Aharon Brodutch (Entangled Networks)
- Topic 5: FAIR workflows
- Topic 6: Workflows for continuum and cross-facility computing

1:30-1:45pm EST

15min Break

1:45-2:00pm EST

Reports from breakout sessions