OAK RIDGE National Laboratory

Automating Science with Workflows at OLCF

Ketan Maheshwari
Sean Wilkinson
Rafael Ferreira da Silva

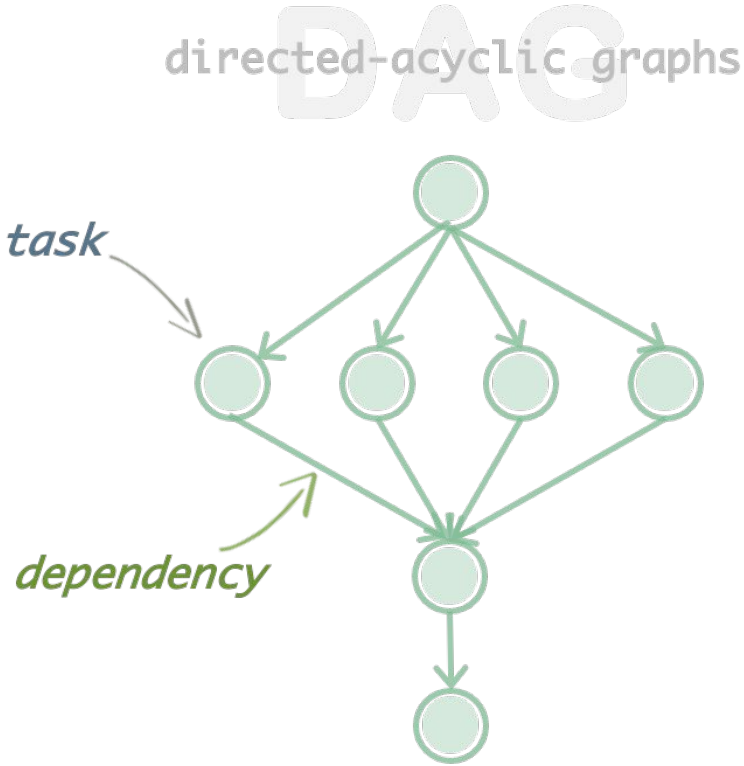
Data Lifecycle & Scalable Workflows
National Center for Computational Sciences

ORNL is managed by UT-Battelle LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

What is involved in an experiment execution?

Scientific Workflows

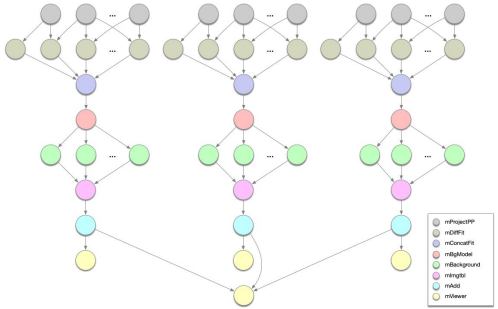


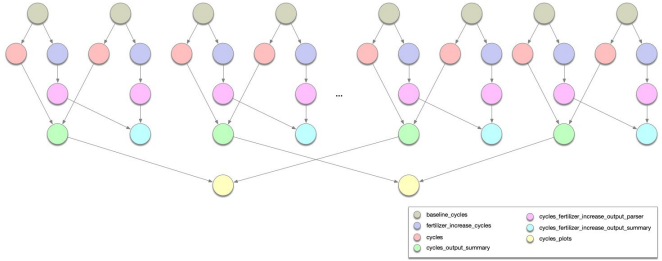
A task often represents a **program** (or script) written in any programming language (**closed box**)

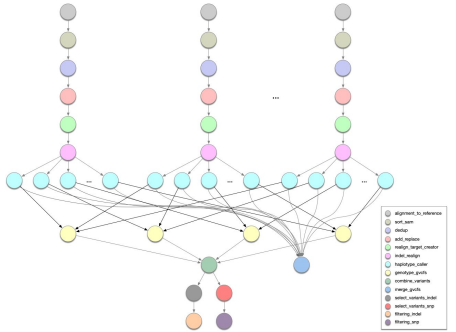
Dependencies are typically based on the **data flow**.

A dependency may also be expressed as **conditions**, **exceptions**, **user triggered action**, etc.

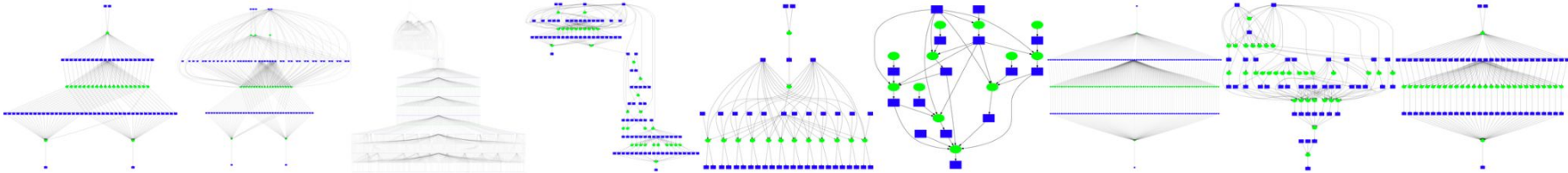
Scientific Workflows







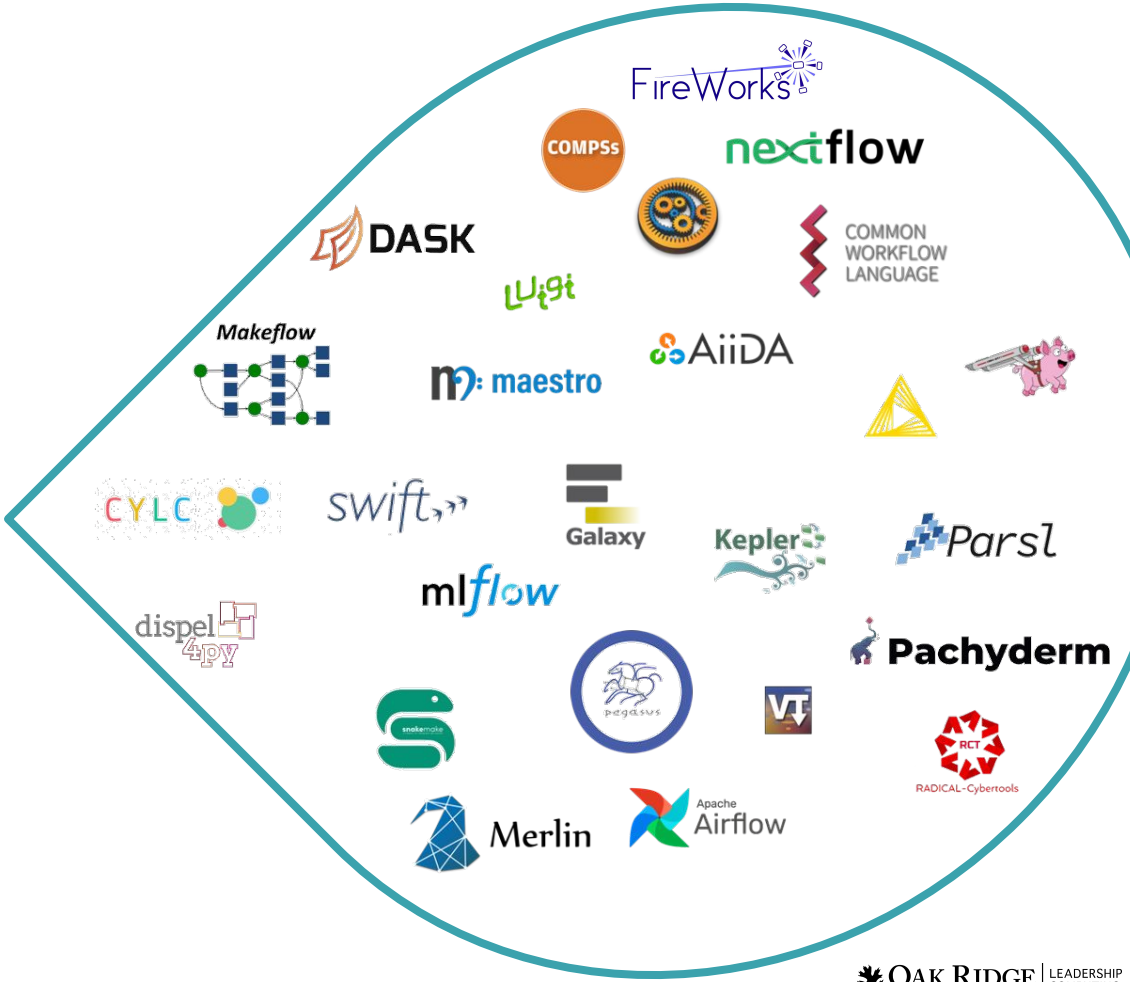
https://github.com/wfcommons/pegasus-instances



https://github.com/cooperative-computing-lab/makeflow-examples

There is a myriad of workflow systems…

The workflow systems landscape is segmented and presents significant **barriers to entry** due to the hundreds of seemingly comparable, yet **incompatible**, systems that exist



https://s.apache.org/existing-workflow-systems
https://github.com/pditommaso/awesome-pipeline

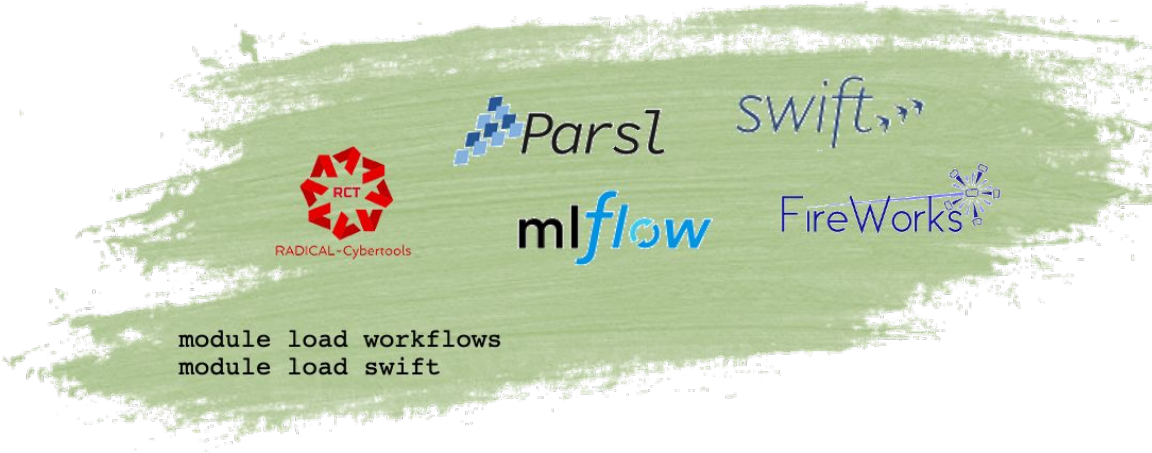
Characterization of Workflow Systems for Extreme-Scale Applications

Workflow Properties	ADIOS	Airavata	Askalon	Bobolang	dispel4py	Fireworks	Galaxy	Kepler	Makeflow	Moteur	Nextflow	Pegasus	Swift	Taverna	Triana
Workflow Execution Models															
Sequential	✔	✔	✔	✘	✘	✔	✔	✔	✔	✔	✘	✔	✔	✔	✔
Concurrent	✘	✘	✘	✔	✔	✘	✘	✘	✘	✘	✔	✘	✔	✘	✘
Iterative	✘	✘	✘	✘	✘	✘	✘	✘	✘	✔	✔	✘	✔	✔	✘
Tightly coupled	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
External steering	✘	✘	✘	✘	✘	✘	✘	✘	✘	✔	✘	✘	✘	✘	✘
Heterogeneous Computing Environments															
Co-location	✘	✘	✘	✔	✔	✔	✘	✘	✘	✘	✔	✔	✔	✘	✘
External location	✘	✔	✔	✘	✘	✘	✔	✔	✔	✔	✘	✔	✔	✔	✔
In situ	✔	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘
Data Access Methods															
Memory	✔	✘	✘	✔	✔	✔	✘	✘	✘	✘	✔	✔	✔	✘	✘
Messages	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Local disk	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Shared file system	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Object store	✘	✘	✘	✘	✘	✘	✔	✘	✘	✘	✘	✔	✘	✔	✘
Other remote storage	✔	✘	✘	✘	✘	✘	✔	✘	✘	✘	✘	✔	✔	✔	✘

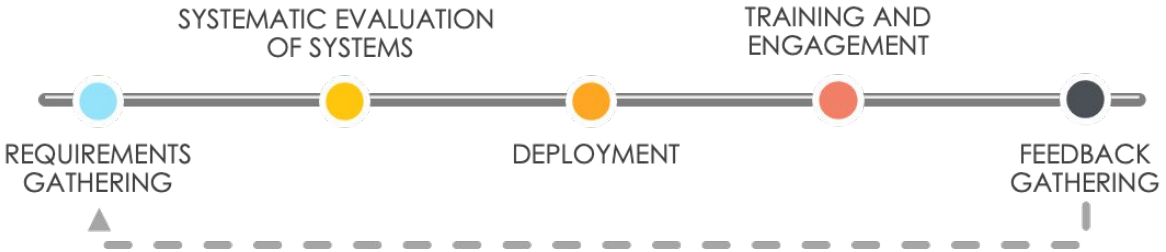
https://doi.org/10.1016/j.future.2017.02.026

Workflows@OLCF

We are constantly evaluating new systems and user requirements, and will deploy them as needed



OLCF's process for deploying workflow systems



Supported workflow systems at OLCF

- Why do we support multiple workflow systems?
  - Why do we support Fortran or Emacs? ;-)
- Important considerations:
  - Documentation
  - User community
  - “Paradigm”
  - Alignment/compatibility with your science and your tools

Ensemble Toolkit (EnTK)

- Python-based
  - Workflows are Python programs that can manage external components.
- Launch workflows from login node with Python.
- Designed for ensemble-based applications such as
  - Molecular dynamics
  - Weather prediction models
- Depends on external services (MongoDB and RabbitMQ)

MLflow

- Library agnostic
  - Workflows are Python, R, Java, or REST API programs that can manage external components.
- Launch workflows by submitting batch scripts.
- Designed with AI/ML workloads in mind
- Can be monitored using built-in web interface
- Lots and lots and lots of provenance tracking

FireWorks

- CLI and Python API
  - Workflows are stored in a database.
  - Workflows can be defined with JSON or YAML files, or they can be Python programs that can manage external components.
- Launch workflows by submitting batch scripts.
- Can be monitored using built-in web interface
- Depends on an external service (MongoDB)

Swift/T

- Swift language
  - Workflows are Swift programs that can manage external components.
- Launch workflows from login node with Swift.
- Designed to take advantage of MPI systems using Turbine and ADLB libraries
- Also available on Crusher and Andes

Parsl

- Python-based
  - Workflows are Python programs that can manage external components.
- Launch workflows from login node with Python.
- Flexible enough to run “anywhere”
- Also available on Crusher and Andes

Workflow Systems Modules on OLCF Summit

Workflow System	Module Command	Documentation / Examples
EnTK	module load workflows entk	radicalentk.readthedocs.io
MLflow	module load workflows mlflow	mlflow.org
FireWorks	module load workflows fireworks	materialsproject.github.io/fireworks
Swift*	module load workflows swift	swift-lang.github.io/swift-t/guide.html
Parsl*	module load workflows parsl	parsl-project.org

OLCF Documentation: docs.olcf.ornl.gov/software/workflows

*ported and tested on Crusher and Andes

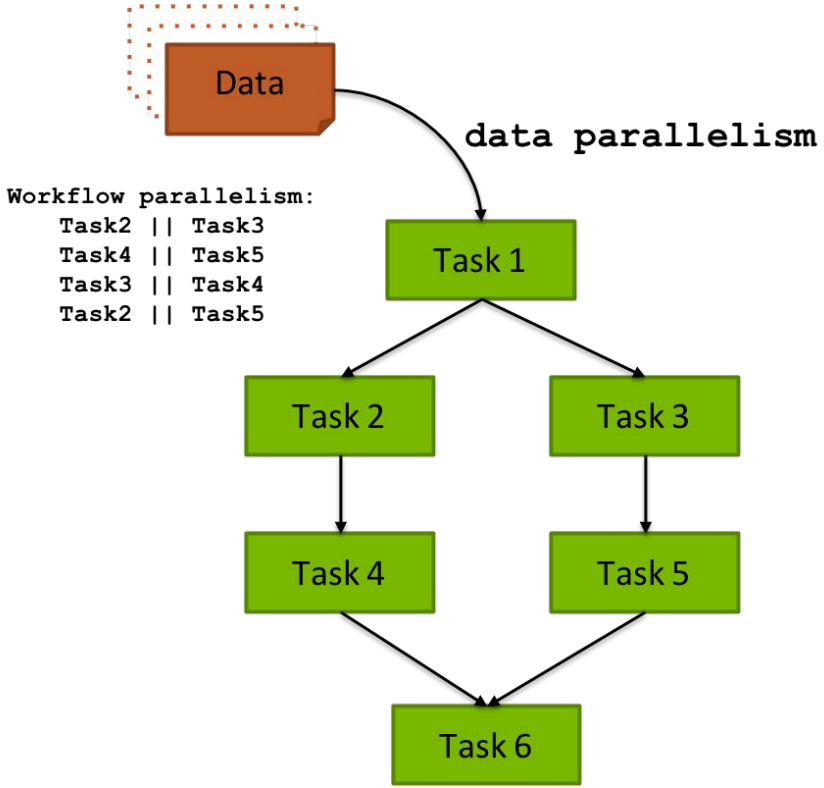
A Quick Demo

Running "Hello World" with Parsl on Summit

Running a hypothetical "Crystal Workflow" with Swift/T on Summit

Running "Hello World" with MLFlow

Running "Hello World" with Fireworks



Get in touch with us

As part of this deployment / support process, we would like to establish **close engagements** with users and applications

We kindly ask you to fill the following form so we can better plan our engagements:

https://tinyurl.com/workflows-olcf

Get in Touch

Community Building

https://workflows.community

managed by a nine-person leadership team, a steering committee, and a technical lead representing

25 companies and institutions from around the world

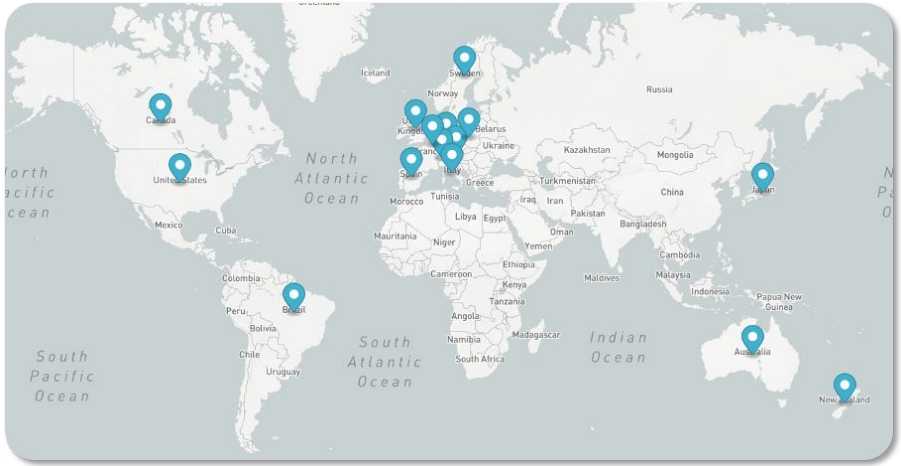
Provide a centralized source for resources, training, workshops, job opportunities, and news to scientists and developers working with workflows

22 workflow systems

99 community members

5 working groups

4 research frameworks



workflows community initiative

Community Summits

Over **100 participants** from a group of international researchers and developers, from **27 workflow systems** and user communities

A Community Roadmap for Scientific Workflows Research and Development

Rafael Ferreira da Silva*†, Henri Casanova‡, Kyle Chard§¶, Ilkay Altintas‖, Rosa M Badia**, Bartosz Balis††, Tainã Coleman†, Frederik Coppens‡‡ˣ, Frank Di Natale[xi], Bjoern Enders[xii], Thomas Fahringer[xii], Rosa Filgueira[xiii], Grigori Fursin[xiv], Daniel Garijo[xv], Carole Goble[xvi], Dorran Howell[xvii], Shantenu Jha[xviii], Daniel S. Katz[xix], Daniel Laney[xi], Ulf Leser[xx], Maciej Malawski††, Kshitij Mehta*, Loïc Pottier†, Jonathan Ozik§¶, J. Luc Peterson[xi], Lavanya Ramakrishnan[xxi], Stian Soiland-Reyes[xvi,xxii], Douglas Thain[xxiii], Matthew Wolf*
*Oak Ridge National Laboratory, Oak Ridge, TN, USA †University of Southern California, Marina Del Rey, CA, USA
‡University of Hawaii, Honolulu, HI, USA §Argonne National Laboratory, Lemont, IL, USA
¶The University of Chicago, Chicago, IL, USA ‖University of California, San Diego, La Jolla, CA, USA
**Barcelona Supercomputing Center, Spain ††AGH University of Science and Technology, Krakow, Poland
‡‡Ghent University, Ghent, Belgium ˣVIB Center for Plant Systems Biology, Belgium
[xi]Lawrence Livermore National Lab, Livermore, CA, USA [xii]University of Innsbruck, Innsbruck, Austria
[xiii]Heriot-Watt University, Edinburgh, UK [xiv]OctoML, USA [xv]Universidad Politécnica de Madrid, Spain
[xvi]The University of Manchester, Manchester, UK [xvii]Tweag, Zürich, Switzerland
[xviii]Brookhaven National Laboratory, Upton, NY, 11973 [xix]University of Illinois at Urbana-Champaign, USA
[xx]Humboldt-Universität zu Berlin, Berlin, Germany [xxi]Lawrence Berkeley National Lab, Berkeley, CA, USA
[xxii]University of Amsterdam, Amsterdam, The Netherlands [xxiii]University of Notre Dame, Indiana, USA

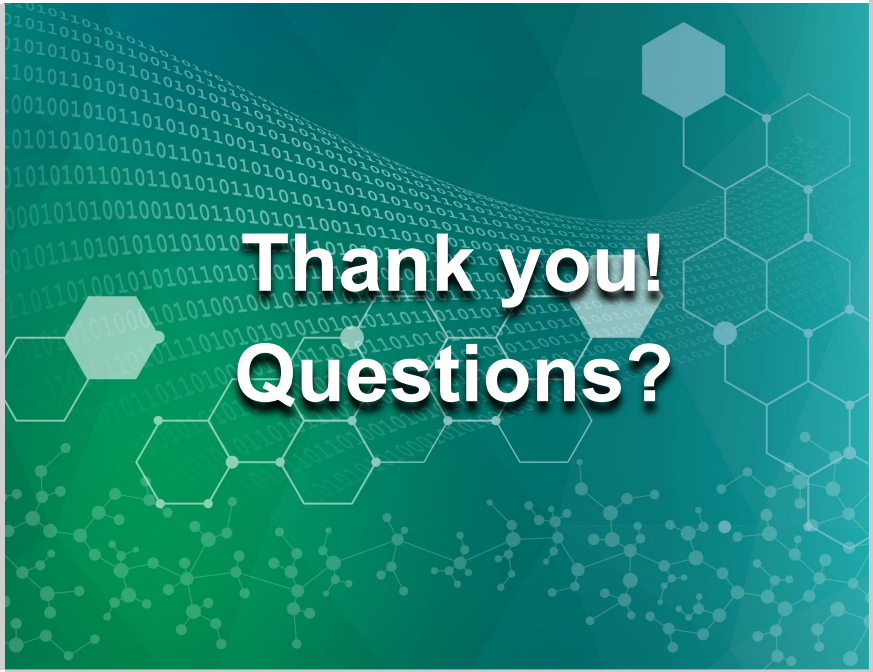
arXiv:2110.02168
https://doi.org/10.1109/WORKS54523.2021.00016

OAK RIDGE National Laboratory

Automating Science with Workflows at OLCF

Ketan Maheshwari
Sean Wilkinson
Rafael Ferreira da Silva

Data Lifecycle & Scalable Workflows
National Center for Computational Sciences



https://tinyurl.com/workflows-olcf

This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.