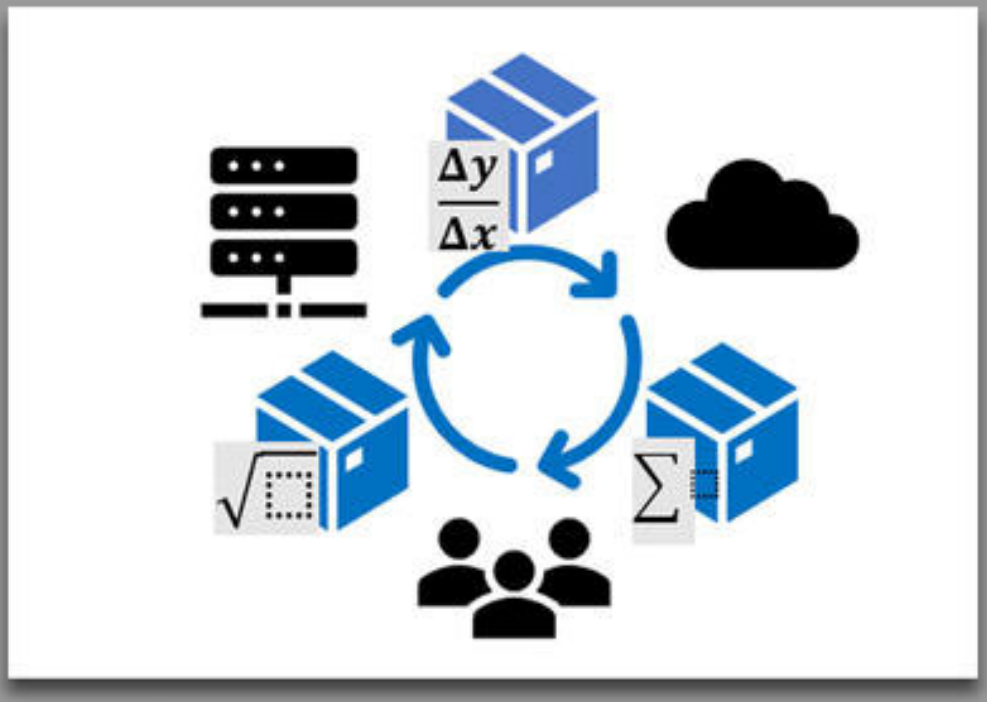
2nd International Workshop on FAIR Containerized Computational Software

December 5-7, 2023



NIST | NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

Welcome to the Day Two of the 2nd International Workshop on FAIR Containerized Computational Software

Co-organized by National Institute of Standards and Technology (NIST) and National Center for Advancing Translational Sciences (NCATS) at National Institutes of Health (NIH)

Peter Bajcsy
NIST

Nathan Hotaling
NCATS NIH

FAIR Containerized Computational Software

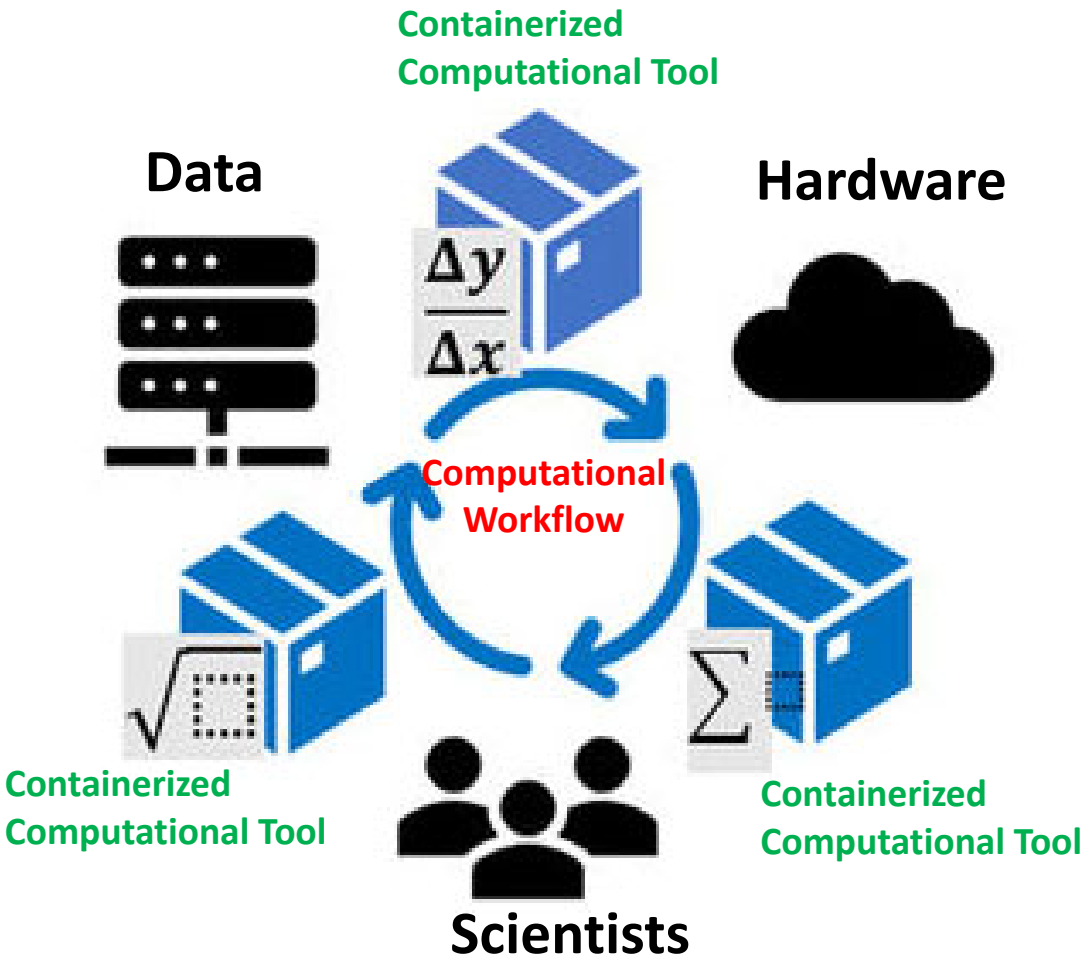
Digital assets: computational software in sciences

Containerization: packaging of software applications so that the software can run in any computational environment

FAIR: Findable, Accessible, Interoperable, and Reusable

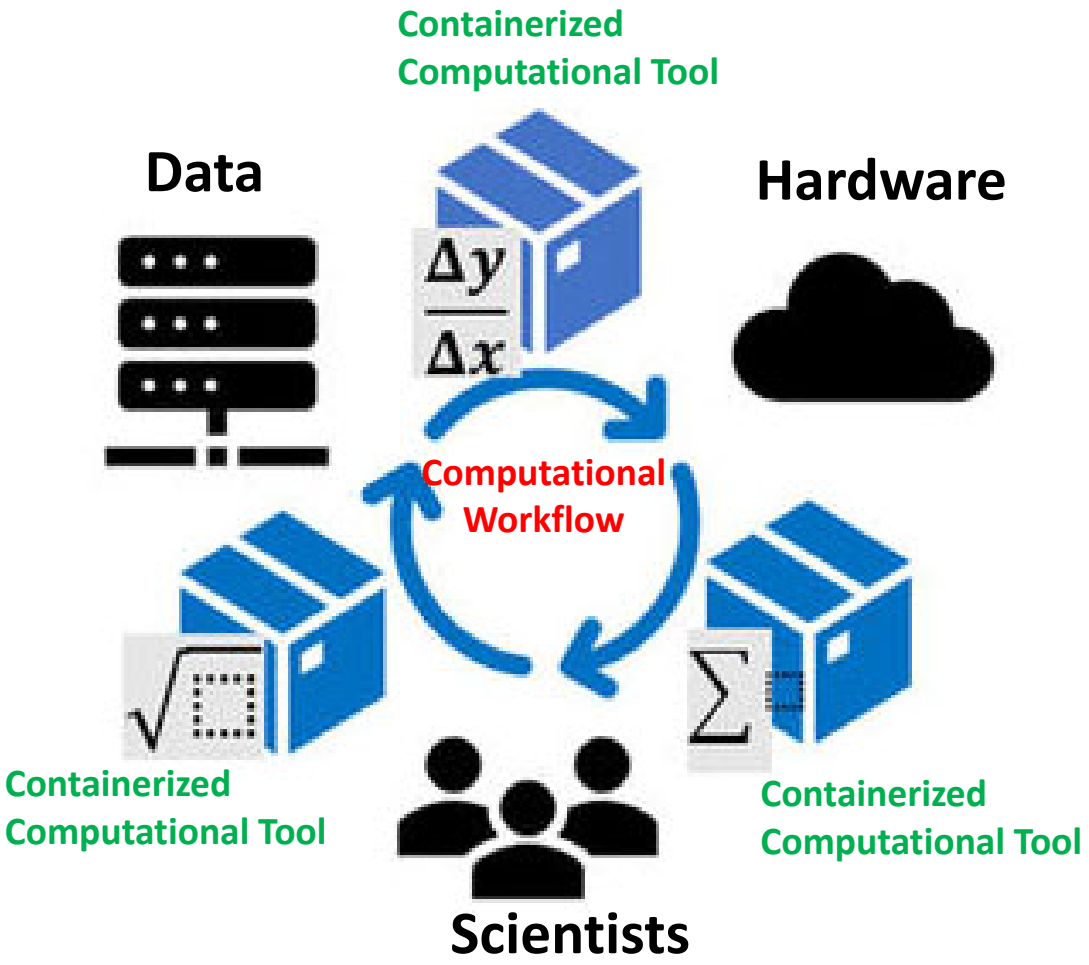
Main goal

The main goal for the workshop is to establish a community consensus on creating interoperable containerized computational tools that can be chained into scientific workflows/pipelines and executed over large image collections regardless of the cloud infrastructure components.



Approach to Forming Computational Workflows

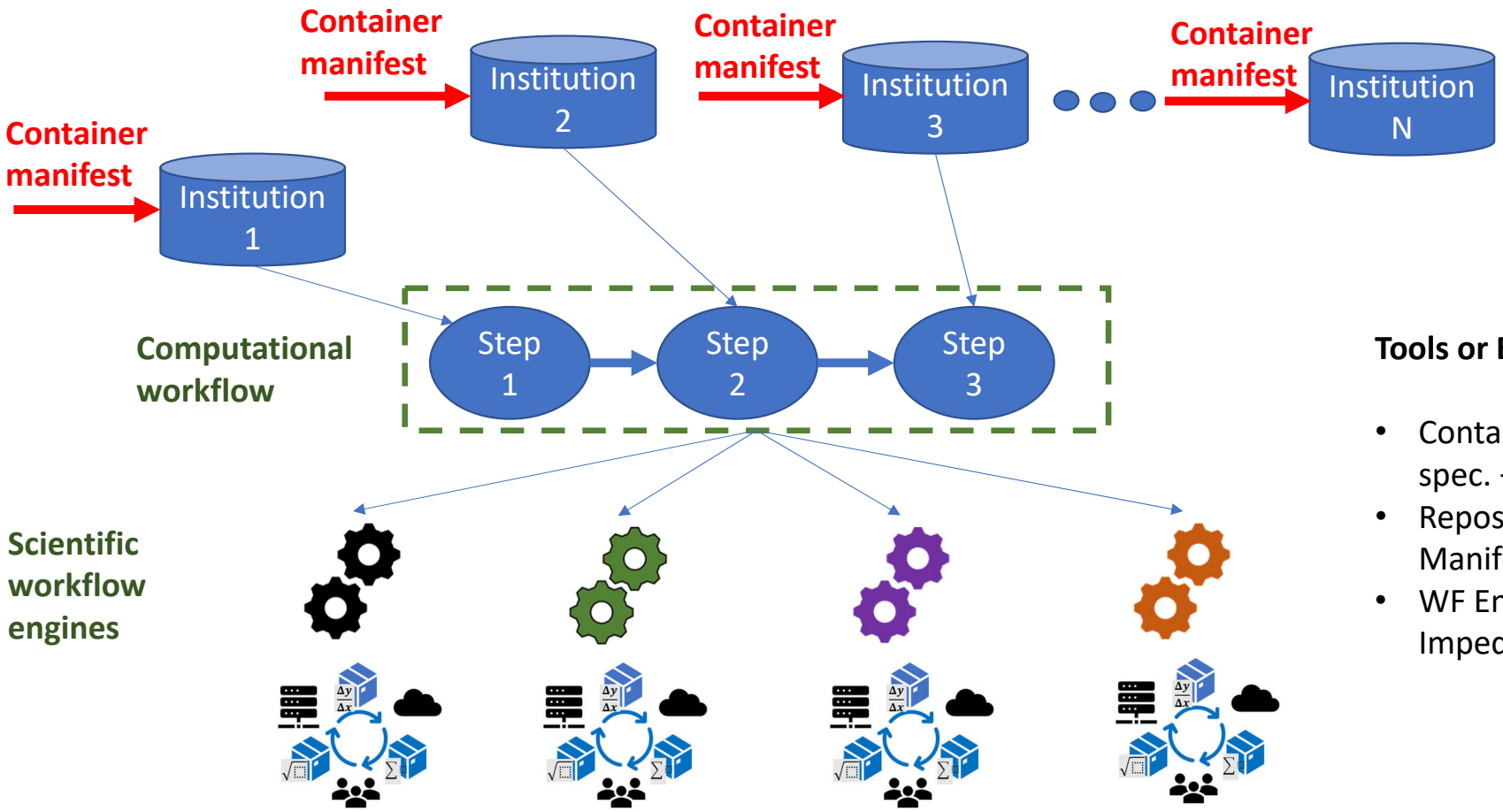
- Create a manifest file accompanying each containerized software tool
- Specify fields in the manifest file for
  - **Inputs/Outputs**
  - **Security**
  - **GUI**
  - **Hardware Requirements**



The purpose of the workshop is:

1. establish fields in a manifest file accompanying each containerized software tool (primary purpose),
2. summarize best practices for containerization of algorithms and interfaces between containerized algorithms and datasets in heterogeneous storage environments,
3. explore application programming interfaces (APIs) for finding containerized software tools and container-based workflows in registries,
4. support executions of container-based workflows in a variety of workflow engines,
5. discuss tooling and governance to support a large community of users.

A Light at the End of the Tunnel



Tools or Eco-system:

- Container manifest spec. + Tools
- Repository with Manifest & WF + API
- WF Engines + SW/HW Impedance Matching

Structure of the 2nd International Workshop on FAIR Containerized Computational Software

Workshop Structure

Themes for each day:

December 5 (Day 1): Inputs/Outputs and Security for FAIR containerized computational software

December 6 (Day 2): Graphical User Interfaces for FAIR containerized computational software

December 7 (Day 3): Hardware Requirements for FAIR containerized computational software

Top level program outline (Each day):

Session 1: General session consisting of opening remarks and background introduction (one hour)

Session 2: Five breakout sessions consisting of about 15-30 people discussing specified topics (two hours)

Session 3: General session consisting of summaries from breakout sessions and closing remarks (one hour)

The Workshop Flow

- Session 1: Main Zoom room
  - After the introduction to the theme of the day, NIST conference facility staff will move participants to breakout sessions to have a uniform representation of organizations across all breakout sessions
- Session 2: Breakout Zoom room
  - Moderators and scribes will go over a set of questions/topics to be discussed. The topics map to the those posted at Federal Registry. The set of questions/topics is the same for all breakouts.
- Session 3: Main Zoom room
  - The moderators and scribes report a set of unique answers/solutions for each question/topic and the results of polls

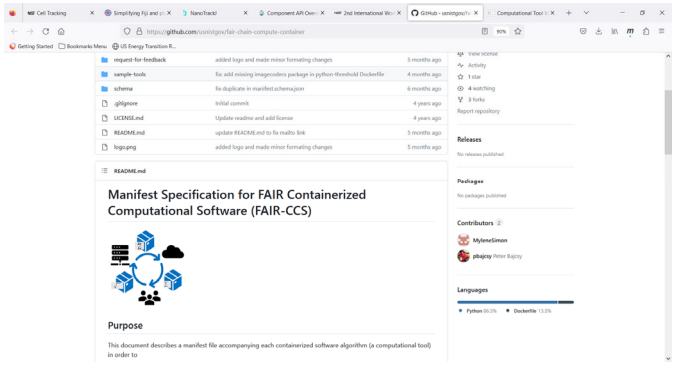
Time and Information Resources

Everyday Meeting Times:

US East Coast (11am-3pm),
US West Coast (8am-noon),
UK (4pm-8pm),
Germany (5pm-9pm),
Korea/Japan (1am-5am)

Comments on the topics outside of the workshop:
https://www.federalregister.gov/documents/2023/08/24/2023-18263/request-for-information-regarding-file-specification-for-findable-accessible-interoperable-and

Workshop event URL: https://www.nist.gov/news-events/events/2023/12/2nd-international-workshop-fair-containerized-computational-software



The GitHub repository with the current specification:
https://github.com/usnistgov/fair-chain-compute-container

Registries of manifests adhering to the current specification:
https://wipp-plugins.nist.gov/
https://wipp-registry.ci.ncats.io/

The Workshop Information

- During the workshop:
  - The workshop is not recorded.
  - The summary notes and chats from breakout sessions will be used for preparing the workshop report.
- After the workshop:
  - Peter Bajcsy (NIST) and Nathan Hotaling (NCATS NIH) will work on the workshop report.
  - The workshop report will be shared with all registered participants for feedback before it will be disseminated to the public.
  - The GitHub repository https://github.com/usnistgov/fair-chain-compute-container will be updated with the workshop report.

Additional Topics

- Use cases
- Ecosystem of tools to support seamless integration to scientific research
- Educational materials
- Benchmarking information
- Governance

Quick Summary of Breakout Sessions from December 5, 2023 (Day 1)

Theme: Inputs/Outputs and Security for FAIR containerized computational software

Quick Summary: 1. What is your experience with generic data organization types?

- **File Locations:** NAS or Cloud Storage - S3 bucket
- **Container ←→ File:** Share files between nodes by storing on disk, serving by HTTP, by persistent volumes or file shares on the cluster through Kubernetes
- **Container Manifest Inputs:** List of paths to input files - Pipeline finds the actual files and brings them (can be a local or cloud path)
- **Container IO code:** Libraries/Tools for images stored in Zarr, OME-NGFF, and Neuroimaging Informatics Technology Initiative (NIfTI) file format

Quick Summary: 2. What is your experience with ontologies for data organization types and file types (availability and usability)?

Pros of ontologies

- Institutes have many different types of users with different roles and ontology could reduce training time for users
- Also such an ontology could be super useful for sharing our containerized components with the outside world

Cons of ontologies

- Not clear whether an ontology would be useful. Maybe for big workflows, or mixing and matching.
- AI for Life tried using pre-defined tags. Had a problem that ontology did not support every tag needed. Not clear if arbitrary tags should be allowed
- Ontology can make it difficult for users to correctly annotate and setup their workflow

Many pointers to ontologies, file catalogs, and frameworks:

- HPC Ontology - https://hpc-fair.github.io/ontology/
- Nf-core - community driven standard- https://nf-co.re/
- S3 Quilt file browser - https://docs.quiltdata.com/catalog/filebrowser
- Bioimage Analysis - visualize & analyze complex images (https://qupath.github.io/)
- List of ontologies used in IDR (https://idr.openmicroscopy.org/about/linked-resources.html)
- OxO - service for finding mappings (or cross-references) between terms from ontologies, vocabularies and coding standards (https://www.ebi.ac.uk/spot/oxo/)
- https://www.ebi.ac.uk/bioimage-archive/rembi-help-overview/; https://www.ebi.ac.uk/empiar/deposition/manual/
- Cryo EM image processing framework (https://scipion.i2pc.es/)
- Pub2tools, RO-Crate: https://www.researchobject.org/ro-crate/
- EDAM bioimaging ontology,
- Fast Healthcare Interoperability Resources (FHIR)
- Brain Imaging Data Structure (BIDS) format
- Huggingface for bioimage models

Quick Summary: 3. What is your experience with data storage that is mounted to a container?

Heterogeneity of storage and access control:

- On premise HPC
- Object store like S3 buckets
- File system behind the firewall
- DVC - Data Version Control
- Role-Based Access Control

Conversions & storage: Conversion of heterogeneous input file into a coherent file format and using cheap storage

- Glacier - slow long-term storage: https://aws.amazon.com/pm/s3-glacier
- IBM Storage Fusion - partial downloading : https://www.ibm.com/products/storage-fusion

Interface with cloud storage:

- Protocols: AI for Life built a server that interfaces with S3, Mounts S3 with open-source server, Creates a policy for each user's home folder, Generates pre-signed URLs to send and receive files in AWS S3
- Nextflow, which handles mounting cloud storages as file systems into the container
- Rembi (images repository supported by riken in Japan; https://ssbd.riken.jp/database/)
- EMBL Embassy cloud – Open stack based (Kubernetes cluster deployment and Ceph Storage backend - https://docs.embassy.ebi.ac.uk/userguide/Embassyv4.html
- Pegasus - Data available on the host OS via NAS or a local filesystem; https://pegasus.isi.edu/documentation/user-guide/containers.html#containers-symlinking; Data from S3 buckets is downloaded to the compute nodes during the data staging phase

Quick Summary: 4. What is your experience with the benefits of including security related metadata?

- **Most participants reported to hands-on experience**
- **Licenses – impose regional, country, file-based license; Security is combined with licensing.**
- **Use Role based access control**
- **Use hashes and signatures against a public-private keys**
- **Protect credentials/licenses especially when doing lot provenance tracking and logging**

Quick Summary: 5. What is your experience with a security-related metadata to verify integrity of container content?

- Security is very relevant when institutions host their own container registries
  - Pull images from authentic source
  - Contain separate registry to check security metadata
  - Don’t want container to connect to the Internet or send information back
- Lessons Learned from:
  - Amazon Elastic Container Registry: https://aws.amazon.com/ecr/
  - Bio Containers: https://github.com/BioContainers
  - **Container digest:** Multiple participants agreed on the usefulness of having a container digest to pinpoint the exact version of the container image and to check its integrity.
  - **Custom hash:** They also agreed that an option for using a custom digest/hash algorithm should be present to avoid relying only on Docker tools.

Quick Summary:

6. What is your experience with encrypting container content?

- Encryption of input data is often more important than encryption of the container themselves

7. What is your experience with protecting a code execution within a container with a passphrase or a license key?

- Instead of encrypting container, control access to location - hardware/systems.
- Role/permission based access/execution

December 6, 2023 (Day 2)

Theme: Graphical User Interfaces (GUI) for FAIR containerized computational software

Agenda

- Session 1: Introduction
  - Peter Bajcsy (NIST) – summary of Day 1 and the goals for Day 2
  - Michael Majurski (NIST) – current GUI fields and the GUI needs at NIST
  - Sunny Yu (NIH) – the GUI needs for collecting algorithmic parameters at NIH.
  - Ben Long (NIST) – registry for manifest files
- Session 2: Breakouts

Date	Breakout	Moderator	Scribe
6-Dec	1	Peter Bajcsy	Kevin Duerr
6-Dec	2	Nathan Hotaling	Guillaume Sousa
6-Dec	3	Michael Majurski	Sameeul Samee
6-Dec	4	Joe Chalfoun	Simo Ouladi
6-Dec	5	Sunny Yu	Philippe Dessauw

- Session 3: Summary