A Community Roadmap to Robust Science in High-throughput Applications

robustscience.org

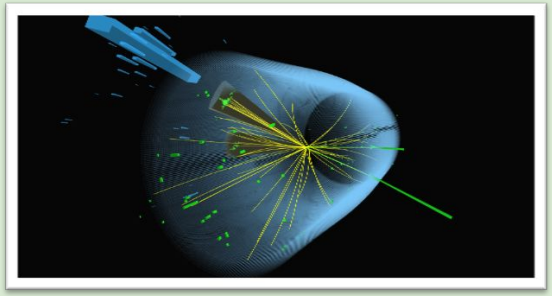
Workflows Interoperability

Rafael Ferreira Da Silva
ISI, USC

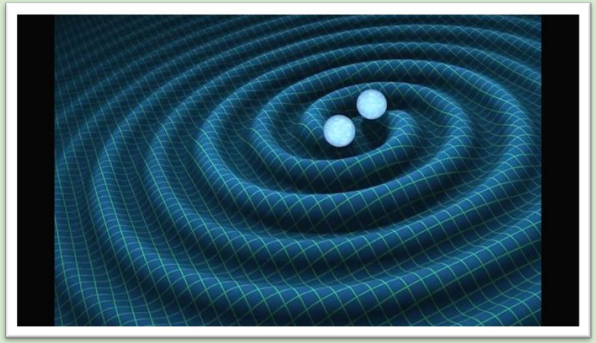
Supported by the National Science Foundation (NSF) under grants number 2028881, 2028923, 2028930, 2028955, and 2028956.

Scientific Workflows

As workflows continue to be adopted by scientific projects and user communities, they are becoming **more complex**



Catering to these workflow features and demands requires **workflow systems research** and **development at several levels**, from algorithms and systems all the way to user interfaces



Workflows are being designed that can analyze **terabyte-scale datasets**, be composed of **millions of individual tasks** that execute for milliseconds up to several hours, process data streams, and process static data in object stores

Workflows Interoperability

To a varying degree, workflow systems implement separately:

- **Language** for wiring workflow inputs/outputs
- Plugin system for connecting possible (often cmd line) **tools**
- **Control** mechanisms (e.g. fault tolerance, loops, choices)
- **Data management** facilities
- Possible **execution backends**
- Ways to connect **multiple workflows** (e.g. nesting)
- **Reproducibility** aspect for sharing workflows
- **Provenance** recording/logging

Workflows Interoperability

There has been an explosion of workflow (orchestration) technologies in the last ten years. Each one serves a different user community or underlying compute engine, albeit with substantial technical and conceptual overlap.

Underlying reasons for divergence:

- Use cases with completely different workflow structures.
- Resources with very different optimization goals.
- Execution systems with fundamentally different capabilities.

There are some missed opportunities for **interoperability**:

- Can user workflows be ported between systems?
- Can provenance be captured in similar ways?
- Can workflow systems be plugged into different engines?

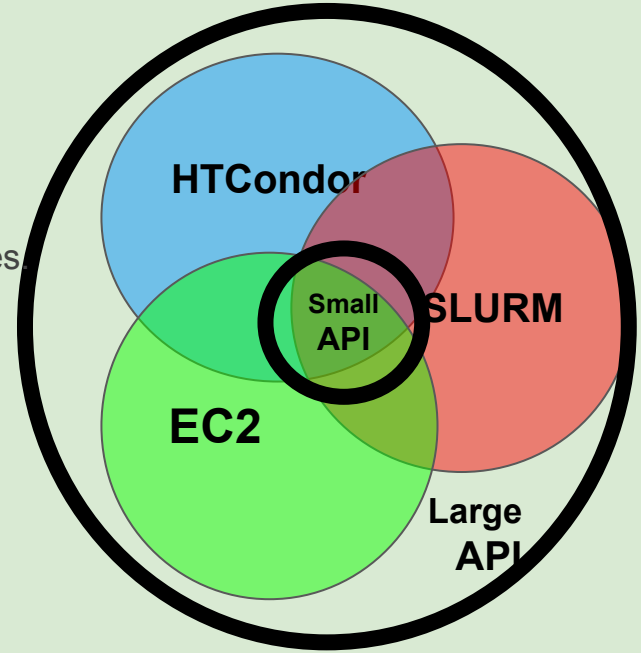
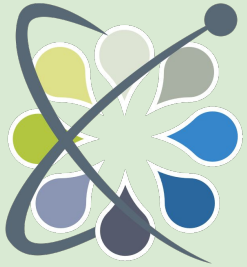


Image courtesy from Douglas Thain (University of Notre Dame) Workflows Community Summit

Workflows Interoperability Efforts



Task execution: DRMAA, GA4GH APIs *(TRS, WES)*, OGF *(JSDL, OGSA)*

Data access: S3**,** DRS, GridFTP, …

Syntactic: Common Workflow Language, Workflow Description Language

Semantic: Bioschemas, EDAM, wfdesc, Biocompute Object, IWIR Metaworkflows, CWFR

Packaging: RO-Crate, BioConda, BioContainers *(Docker/Singularity)*, Debian-Med

Data: HDF5, VOTable, CSV On the Web, SBML (COMBINE), HL7 FHIR, DFDL

Metadata: DCAT2, Codemeta, Datacite, schema.org, W3C PROV

Repository: WorkflowHub.eu, Dockstore, nf-core, PegasusHub, Galaxy Toolshed, bio.tools

Platforms and Hardware: HTC, HPC, OpenCL, Rosetta 2

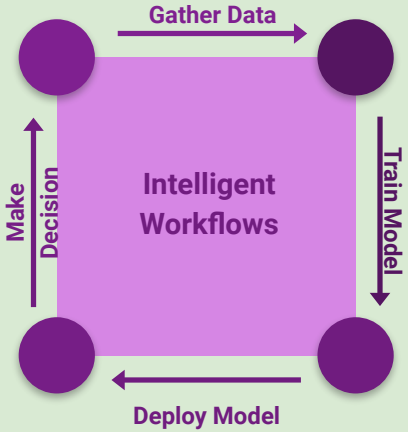
Slide courtesy of Stian SoilandReys (University of Manchester)
Workflows Community Summit

AI Workflows

AI Workflows have artificial intelligence / machine learning systems as workflow components, sub-components and/or orchestrators.

Interoperability is even harder due to the following **challenges:**

- **Data driven** computing
- **Heterogeneity** in software *and* hardware
- Distributed, edge and high-frequency computing
- Simulation results are not necessarily the product
- Leveraging fast-paced developments, driven by industry, not science



Slide courtesy of Luc Peterson (LLNL)
Workflows Community Summit

FEEDBACK OPPORTUNITIES

This project needs your feedback and there are 3 ways that you can contribute:

- **No. 1 Before the Cafe:** On the website there is a form available, per cafe, where you can provide use cases for the session in relation to the topic question.

- **No. 2 At the Cafe:** Contribute to the MURAL design-thinking application tool during and after the session.

- **No. 3 After the Cafe:** Complete the post-workshop survey designed by the Assessment Team.

Visit the project website at: **robustscience.org**