

A common workflow registry of compute endpoints and applications

Daniel S. Katz (UIUC), Kyle Chard (ANL/UC), Rosa M. Badia (BSC), Jorge Ejarque (BSC),

There is now a large and ever-growing number of workflow systems, and we have lost hope in encouraging users not to continue developing more. Instead, we want to focus on building shared elements that can help us with our own systems, as well as the users of those systems and the developers of applications that will increasingly be used as workflow elements in simulation, analysis, search, optimization, and parameter study research campaigns.

Two of the common types of elements that workflow systems interact with are the end computing systems and the preexisting applications that the workflows wrap and call. Today, users of a workflow system have to find information about both the end points and the applications, they have to map that information to workflow-specific configuration formats, individually customize their workflow to use these configurations, and keep up with changes over time. Instead, we propose a registry of compute endpoints and applications, where an entry could be automatically brought into a workflow system.

This requires a pair of components:

- The registry itself and a means for adding and editing entries, potentially along with curation, or perhaps community curated, using WikiData
- A means to use entries for a given workflow system

Registry entries could be added by three different groups:

1. Compute resource providers could enter their systems, and application developers could enter their applications
2. Workflow system providers could enter systems and applications that they support, or we could collect published configurations and map them to our common schema
3. Workflow developers could enter system and applications that they and their workflow users need

We would like to create a project/collaboration that would build:

- A prototype of the registry itself and a means for adding and editing entries, potentially along with curation, or perhaps community curated, using WikiData
- Methods to use registry entries in Parsl and PyCOMPs

We believe the initial work is:

1. Defining the schema for the registry, and implementing it as a REST service

2. Building some test elements, and entering them manually
3. Building software for Parsl and PyCOMPs to import and use registry entries

We had planned to do this initial work in the summer of 2020, but Covid-19 has delayed a kickoff meeting we planned in the spring, and has decreased our ability to do extra work on our own. We are writing this whitepaper to see if there is interest in this idea from others.

After this initial work, we will evaluate progress and decide on next steps, which likely include a publication and bringing in additional workflow systems.

The results we would initially work towards are:

- A paper describing the prototype
- The prototype registry
- Initial methods added to Parsl and PyCOMPs to use the registry

Are you interested in this idea? If so, let's coordinate and get started!