Framework for workflows with task dependencies on HPC

Scott Zimmerman, MPH, a researcher and software developer working with Chancellor's Professor of Public Health Jennifer Ahern, will present on his solution to migration of a workflow that includes interdependencies between tasks, from Amazon Web Services to the campus's shared HPC cluster, Savio. Prof. Ahern's group studies and develops methods for social epidemiology, especially related to the links between violence and health. Scott develops simulation software to advance this work, comparing study design and analysis methods in terms of bias and other metrics.

As Scott describes the framework he has developed: "Some computational jobs benefit from division into small tasks, allowing for simplified debugging, logging, and error management. Sometimes tasks are related to each other via complex dependency structures that are not known prior to beginning computation, requiring the use of scheduling to coordinate tasks. Processes running in parallel can handle the tasks as their dependencies are met by using a database to coordinate tasks. I used this strategy for AWS-based workflows, but upon transitioning to Savio-based workflows I found that latency severely hindered performance when the database was out-of-network. Furthermore, use of a database incurred extra costs and added complexity to our architecture. To solve these problems I developed a simple task scheduler that uses socket communication to coordinate tasks. In parallel jobs, the task scheduler runs as a server on a single node, and compute processes communicate with the scheduler via a client. In this presentation I will discuss the motivating problem, the resulting scheduler software, and potential use cases."

When: Thursday, May 18, 2017 from 12 - 1pm
Where: 200C Warren Hall, 2195 Hearst St (see building access instructions on the parent page)
What: Framework for workflows with task dependencies on HPC
Presenting: Scott Zimmerman (School of Public Health)

Before the reading group, please review the following materials:

- The readme (and, optionally, code) for the TaskScheduler developed by Scott, on GitHub
- Interested attendees who are not familiar may also want to familiarize themselves with Google's protocol buffers, which the scheduler uses for message serialization
PS: Can controller use hybrid resources, e.g., Savio & Cloud
SZ: So long as access to storage and communication with participating resources is there, yes. Latency could prove a problem. Data might want to be on S3 because scratch can't be read from outside the Savio cluster.
PS: Multiple queue access might be of interest too ... low-priority for small tasks, big tasks for condo might be advantageous for some users. Policy for where to spawn jobs might be of interest.

AC: On use cases, how does SZ conceive the generalizability of this software
SZ: Created to solve a problem I was facing; discussion with Krishna led to this reading group, considering whether and how this could be developed for more general use.

MM: Adesnik "round trip" experiment ... there’s a head-node, doing some management and capabilities. Similarity there.
PS: This software's respawning ability works in approximately this way. I wonder about Jupyter notebooks working in this way, to spawn jobs into Savio.
MM: Jupyter -- for visualizing results as well. Perhaps an interesting way to think about extending/generalizing.

SZ: Open to contributions/ideas. Message me on GitHub. Current goals: to finish additions on current roadmap, make it available on GitHub
MM: Would be happy to demonstrate Jupyter spawning into Savio
SZ: That'd be great...