15th JLESC Workshop

Joint Laboratory for Extreme-Scale Computing

Contribution ID: **73**

Type: **Short talk**

Data analysis, interactive development, and the Julia Language with HPC Distributed Systems.

Thursday 23 March 2023 10:50 (10 minutes)

Data is messy. What's more, the most tantalizing data to study is often that which is new and has not attracted attention yet. This tends to be the messiest. One of the major driving forces in the popularity of interactive programming is the ability to be flexible with an unknown data-space. Environments such as Jupyter Notebooks have become ubiquitous in data analysis for this reason. And rightfully so. They allow a developer to rapidly build up code to adapt without having to recompile-reload program and data from scratch. But it often comes at a cost. For example, Python is notoriously slow in a variety of ways and resources allocated go unused while waiting for a developer to enter their instructions. Initially it is unclear if the need for rapid and serendipitous code development outweighs the need for raw processing speed and efficiency. Experience has shown that for desktop computing and data analysis, indeed, interactive development reigns supreme.

For a variety of practical reasons, HPC systems tend to employ a batch scheduled computing model. While this should be the major mode of operation, more space should be made for interactive development with distributed computing to replicate the success of interactive development seen in desktop computing. JuliaLang is a good candidate to fill this space and expand past it. Julia is interactive with a JIT compiler, inherently asynchronous (e.g. multithreading and more), and has varying degrees between dynamically and statically typed. Julia programs can be interactively and dynamically developed, optimized, and scaled to approach performance of a traditionally compiled program.

In this short talk I will start by describing high level features that make Julia an ideal candidate for interactive distributed computing. Then I will introduce a specific problem that involves a sufficiently messy dataset. It is too large for AI/ML analysis on a desktop computing enviroment. For this problem I've developed and will introduce two Julia packages, DistributedQuery, DistributedFluxML, that help with in memory distributed data hosting and distribute AI/ML training. I will finish the talk seeking collaboration to improve, harden, and find more novel data analysis problems that can capitalize on an interactive and distributed development environment.

JLESC topic

Primary author: SAXTON, Aaron (NCSA)

Presenter: SAXTON, Aaron (NCSA)

Session Classification: Short Talks on Interative Tools and Monitoring

Track Classification: Programming languages and runtimes