Developing a Website for High-Level Visualization of Modulefile Usage Data
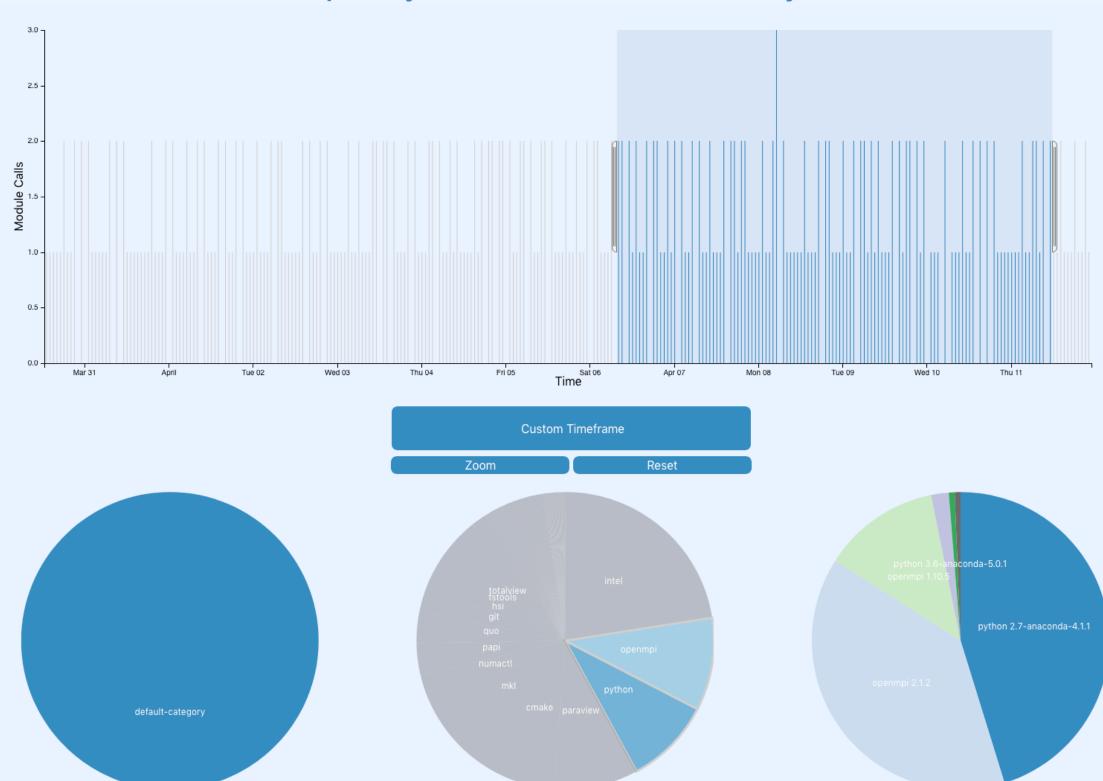
Trent Steen (Dakota State University)

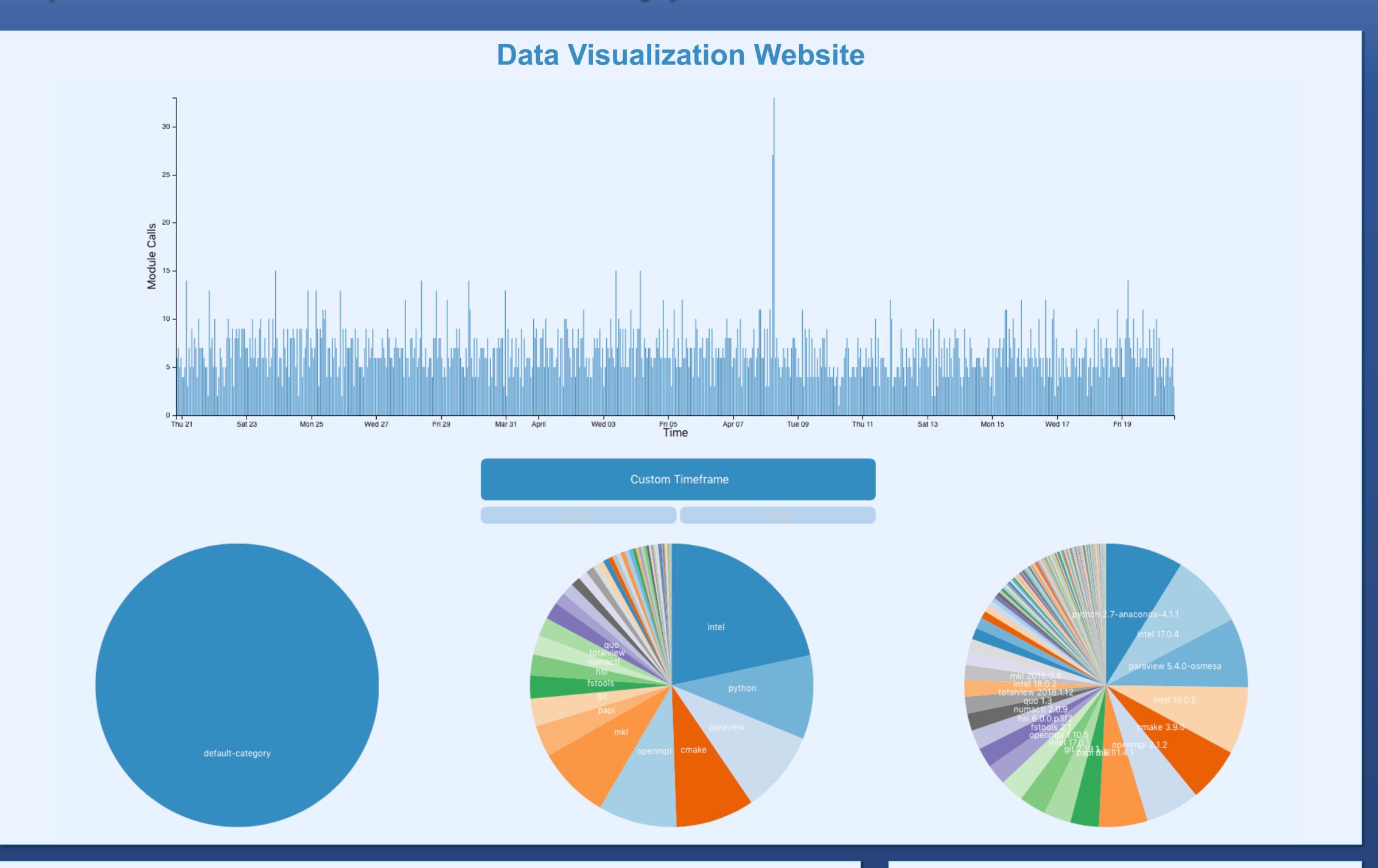Los Alamos National Laboratory (LANL)'s Advanced Simulation and Computing Program spends about $2.5 million annually maintaining and licensing scientific software on production HPC resources. A significant proportion of this cost is consumed ensuring that required combinations of installations are available. In an effort to reduce the support overhead required in maintaining the full combinatorial explosion of installations for a versatile runtime environment, it's paramount to monitor modulefile utilization data and generate dependency lists to target specific permutations, while ignoring those identified as unused.

Empirical analysis of LANL HPC's modulefile usage data indexed by Splunk describes specific software dependency combinations most often loaded by customers. This work aims to demonstrate an external interface for stakeholders of HPC that serves both as a layer of abstraction from sensitive data and as a more practical tool for generating targeted reports for specific audiences. The knowledge derived from this tool will help conserve resources and mitigate problems stemming from brute-force over-provisioning by applying informed constraints to software toolchains and enabling the software team to focus on real customer requirements at LANL.

Data Visualization Website

Site Features

Data is sorted by category, product, and version, and displayed in three corresponding pie charts. Users can click on any slice to filter all charts by data from that slice, or click and drag on the bar chart to specify a timeframe to filter by.

Client Side vs. Server Side

For this first version of the website, the data is sourced directly from a usage log stored on the yellow network. A Python script cleans this up and randomly selects 0.1% of 2019's data to be loaded directly into the web browser, where it's manipulated and displayed using the dc.js library. In order to use the entirety of Splunk's logs in the final version, data manipulation will need to be done server side. The web browser will send a request to an intermediary server containing pre-coded Splunk credentials and searches; this server will then request data from Splunk and send it to the browser to be displayed.

Future Work

- Getting permission to host site and intermediary server on network
- Obtaining Splunk credentials
- Writing Splunk searches to populate site data
- Reworking user interface to account for having to send multiple requests to the server during any given session
- Categorizing products for the leftmost pie chart so its filter can be used to declutter other two pie charts
- Potentially implementing a regex feature that will allow users to create more customized searches

Los Alamos
NATIONAL LABORATORY
EST. 1943