CS486C – Senior Capstone Design in Computer Science

Project Description

Project Title: A tool to help climate scientists containerize their code	
Sponsor Information:	Nicholas McKay, Associate Professor, Paleoclimate Dynamics Laboratory School of Earth and Sustainability, NAU nick@nau.edu 928-606-9823

Project Overview:

Climate change is a defining challenge of the 21st century, and climate has always been fundamentally tied to societies and environments. To forecast and prepare for climate change, it is fundamental that we understand how and why climate has changed in the past. The study of past climates, or paleoclimatology, deals with these questions, and scientists travel the world to understand how and why climate has changed in both the recent, and distant, past. To do this, we study tree rings in the Southwest, ice cores in Greenland, corals in Bermuda, and lake mud in the Arctic. Each of these studies provide small snapshots into what climate looked like in these places hundreds or thousands of years ago.

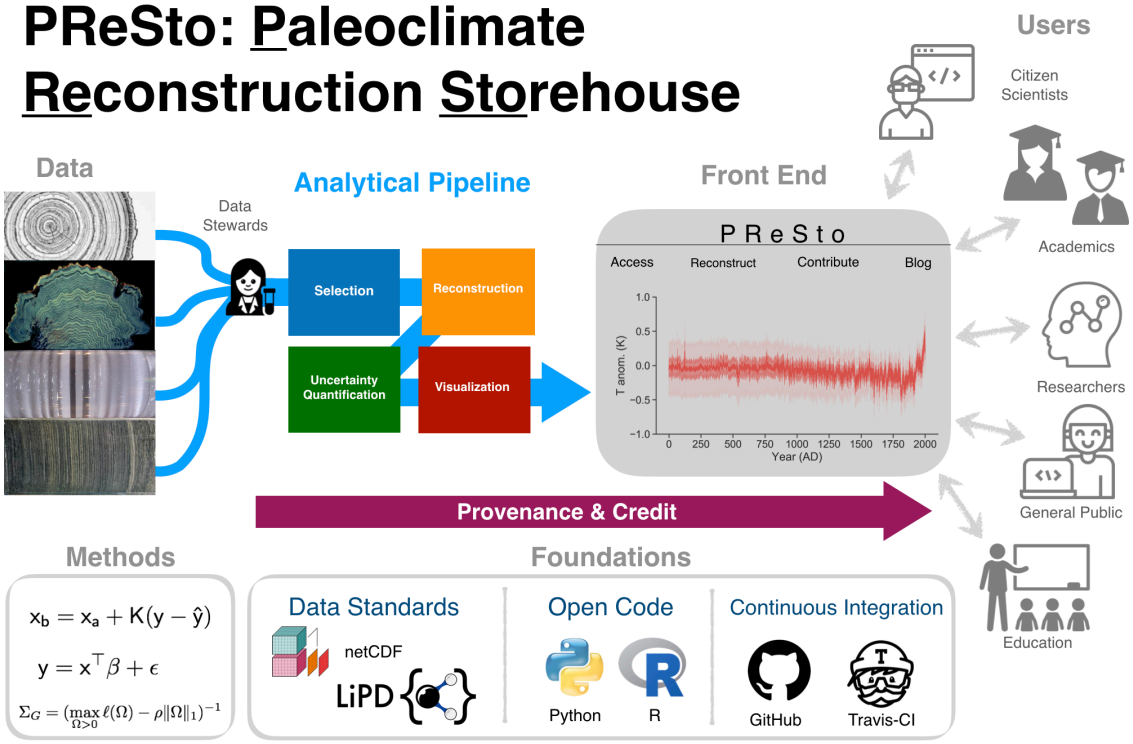

Figure 1. Conceptual design of PReSto. This project focuses on helping scientists contribute methods to the analytical pipeline.

Over the past 30 years or so, paleoclimatologists have now developed thousands of these paleoclimate datasets from around the world, and major international efforts now focus on integrating all of these separate datasets into unified "climate reconstructions", effectively maps that show how temperature or precipitation for specific regions at various times through Earth's history. Traditionally, these efforts only happen sporadically, and become quickly outdated. To address this, the Paleoclimate Dynamics Laboratory at NAU, along with collaborators at the University

of Southern California, have launching a project called PReSto (Paleoclimate Reconstruction Storehouse), that will streamline this process, and keep climate reconstructions up to date with the latest datasets and most state-of-the-art methods.

A major challenge with this effort is that the scientists who develop the code to produce these reconstructions use an array of programming languages, libraries and community-supported packages, meaning that it can be exceedingly difficult to run these reconstructions on different computers, especially as time goes on. This fundamentally limits reproducibility. Recently, the advent of containerization approaches (e.g. Docker, Kubernetes, Singularity), has provided an efficient mechanism to solve this problem, however few scientists have experience with such solutions. That said, some recent efforts have developed tools for automatic containerization that could be building blocks for this project.

Use case:

Cindy, a climate scientist, has written a lot of code in Python that takes input data and parameters, runs it through a sophisticated algorithm, and produces output files that show estimates of climate over the past 2000 years. Now, she'd like to contribute this code to PReSto, so that her reconstructions can be automatically updated, and readily used by other scientists. She knows about containerization, but not the details of how to contain her code to make it work with PReSto. This project will produce a software solution that will walk her through the steps of creating a PReSto-ready container, by parsing her code, identifying key inputs, parameters and outputs, asking for input and documentation where needed, and testing the final container for suitability, while providing efficient avenues for updating and maintaining the codebase and containers.

The solution

We envision this as a GUI (possibly as a secure web application) with a high-quality UI/UX that builds on existing components to streamline and simplify the containerization process. Some of the key features would include.

- A straightforward interface that would guide programming-literate scientists through the steps of containerizing their code, including downloading an image or creating an image of their own.
- A language agnostic solution (although most code sources will be in python or R, and all will be from open-source languages).
- The capacity to examine source code and help identify inputs, outputs and parameters, and ask the user to describe the requirements of each of these characteristics
- Implementation of unit tests to evaluate successful containerization.
- Integration of the solution with individual accounts on platforms such as GitHub for code and DockerHub for containers so scientists can update their code/containers as needed.

If successful, this solution could greatly expand the number of scientists that could contribute to PReSto, and the breadth of adoption of their methodologies. Few scientists are comfortable implementing containerization thus far, which significantly limits the reproducibility of this aspect of climate science.

Knowledge, skills, and expertise required for this project:

- Knowledge of modern web2.0 design and development tools and techniques.
- UI/UX: knowledge of web-app capabilities, constraints, and frameworks for web GUI construction.
- Containerization software and implementation

Equipment Requirements:

- There should be no equipment or software required other than a development platform and software/tools freely available online.

Software and other Deliverables:

- A fully functional solution, installed and demonstrated on platform of our choice.
- A strong as-built report detailing the design and implementation of the product in a complete, clear and professional manner. This document should provide a strong basis for future development of the product.
- Complete a professionally-documented codebase, delivered both as a repository in GitHub, and as a physical archive on a USB drive.
- A working solution that we can workshop with scientists interested in contributing methodologies in April 2022.