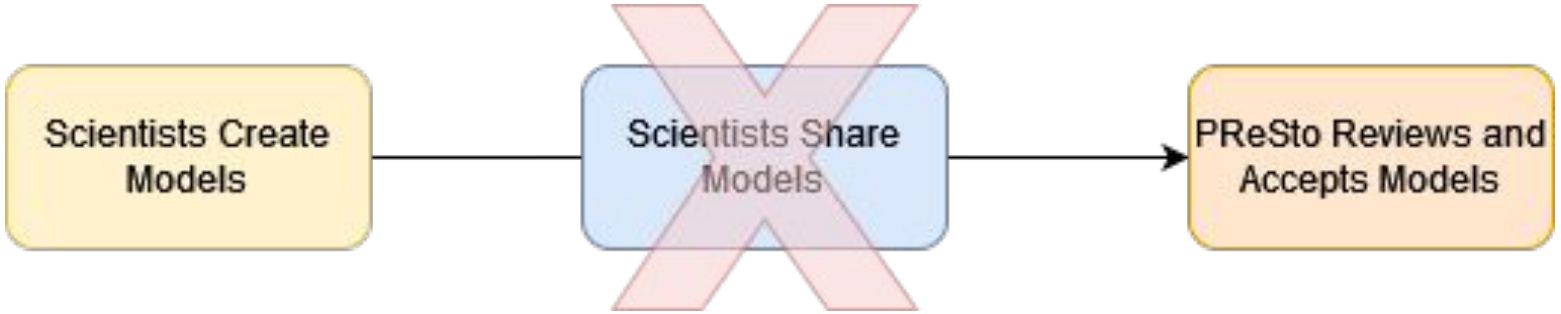
Design Review I

Fossilized Containers: Emily Ramirez Serrano,
Jeremy Klein, Jadon Fowler, and Mumbi Mbuthia

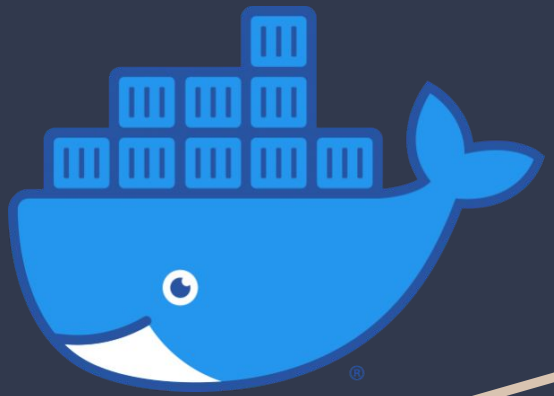
Background

- Paleoclimatology
- Paleoclimate reconstructions or **PRs** or models
- Creates representations of climates
- Paleoclimate Reconstruction Storehouse or **PReSto**
- Center for PRs developed by Dr. McKay



Workflow Diagram

Problem Statement



- Containers!
- Container = interactive snapshot of a computer system
- Now the problems are:
  - Create containers of PRs
  - Guide paleoclimatologists through containerization of PRs
  - Scan PRs for input, output, and parameters

Solution Vision

PReSto Containers

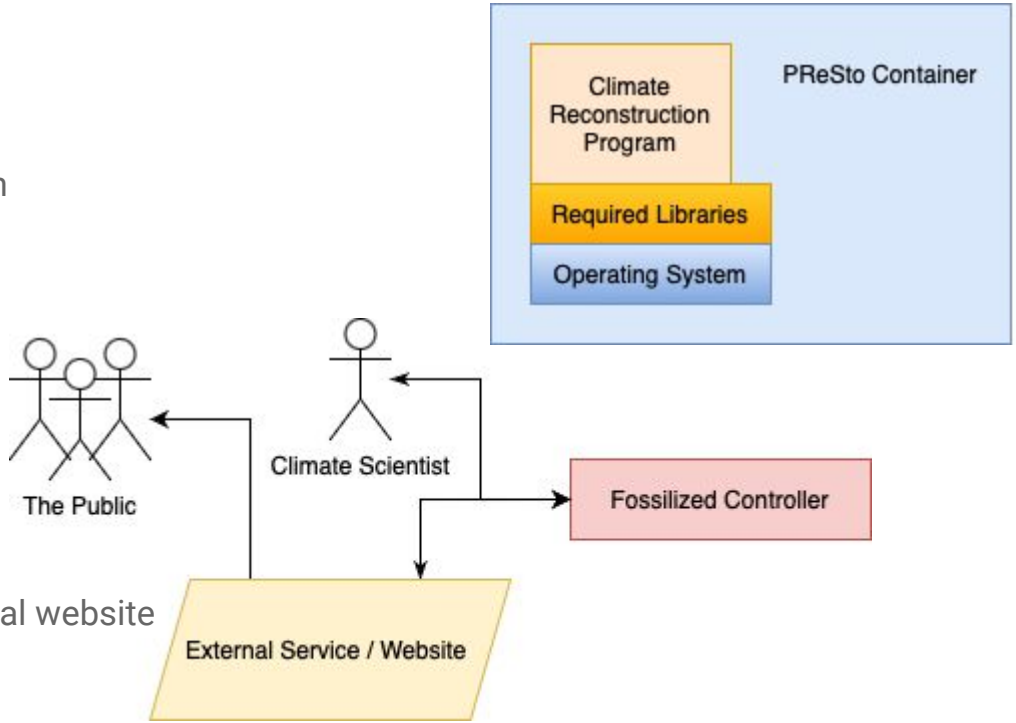
- Containerized Climate Reconstruction Program

Fossilized Controller

- User Interface for building PReSto Containers

External Services

- PReSto Containers can be viewed on an external website

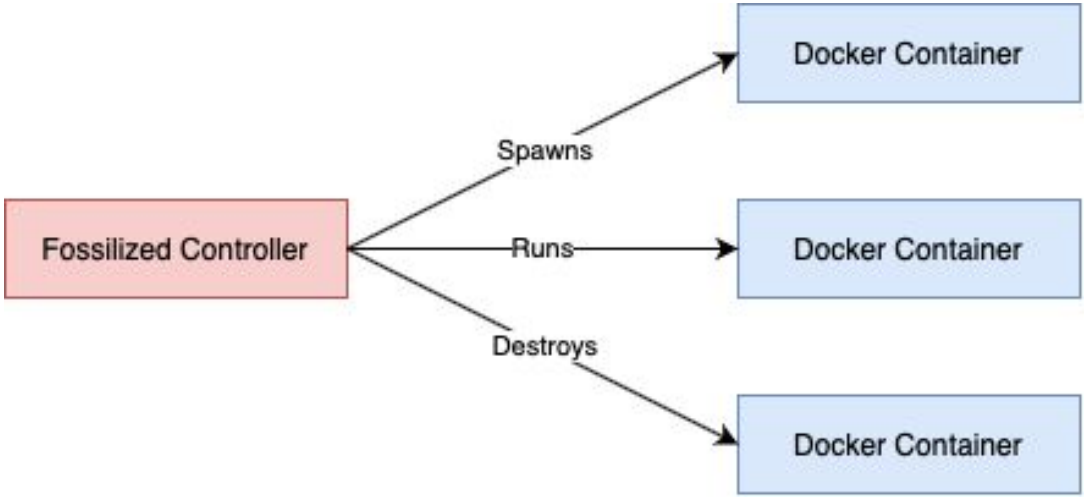


Key Requirements

- Requirements were obtained in weekly meetings over Zoom
- Fossilized Controller contains a **CLI tool**
- PReSto Containers are **Docker Containers** that communicate with the Fossilized Controller
- Documentation for the entire process will be hosted online

Requirement 1: Fossilized Controller is simple to use.

Anyone can read the documentation online for our Command Line Interface (CLI)!

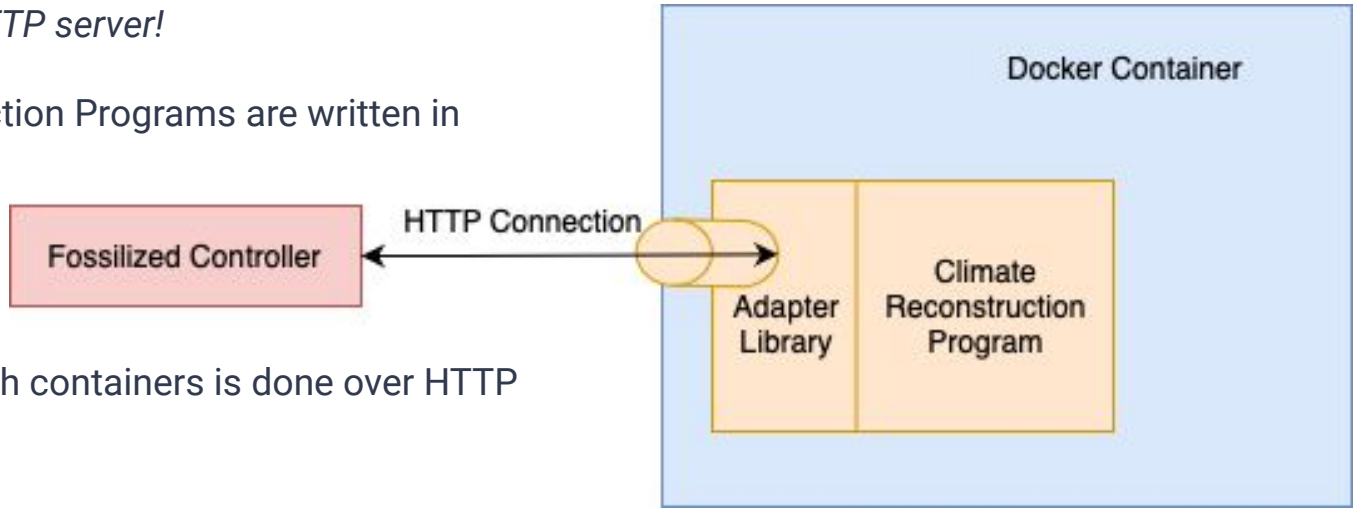


Key Requirements

Requirement 2: our solution is language agnostic.

Every language has an HTTP server!

- Climate Reconstruction Programs are written in many languages



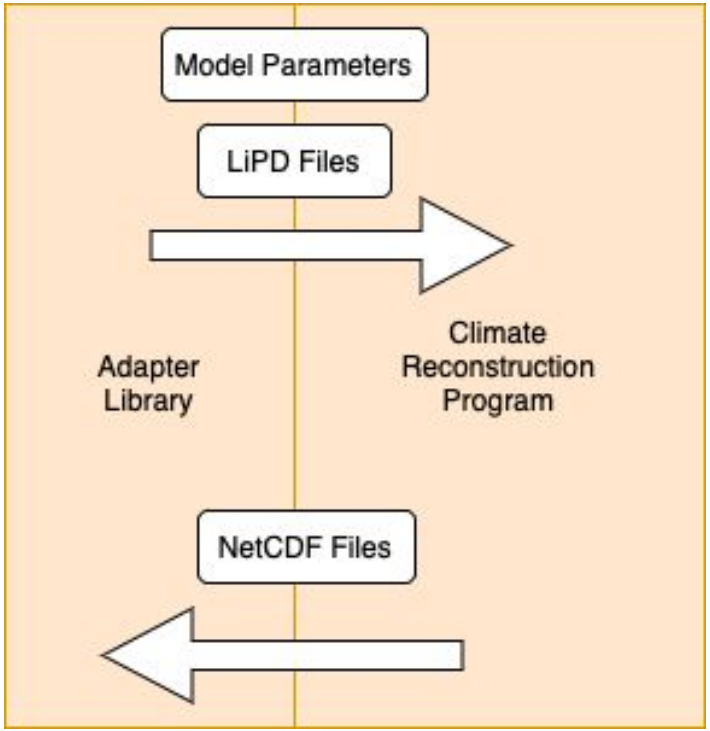
- Communication with containers is done over HTTP

Key Requirements

Requirement 3: quick development with Python and R

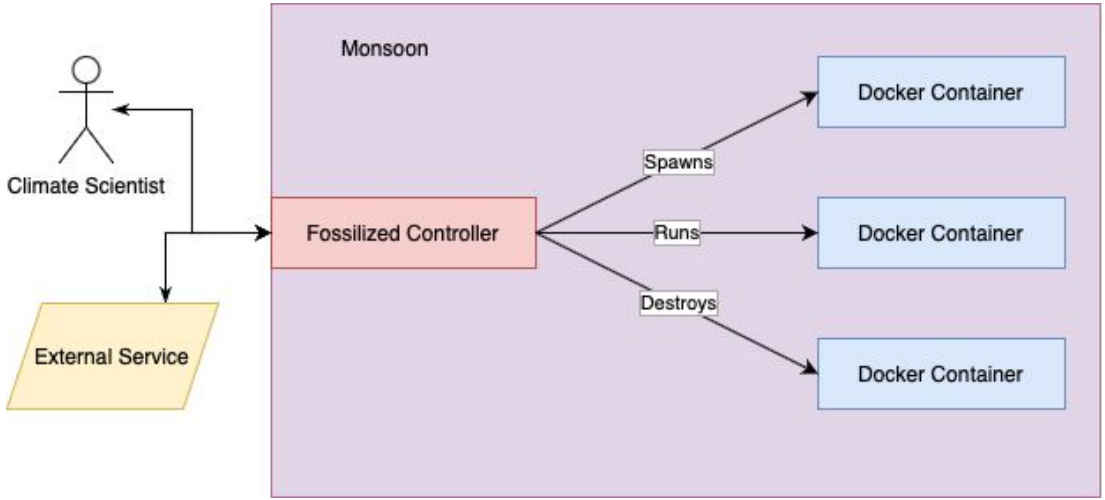
We need Python and R packages!

- Linked Paleo Data (**LiPD** files) and Model Parameters are sent by the **Adapter Libraries**.
- The resulting reconstruction is in Network Common Data Form (**NetCDF**) is given to the Adapter Libraries to send to the Fossilized Controller.



Performance & Environmental Requirements

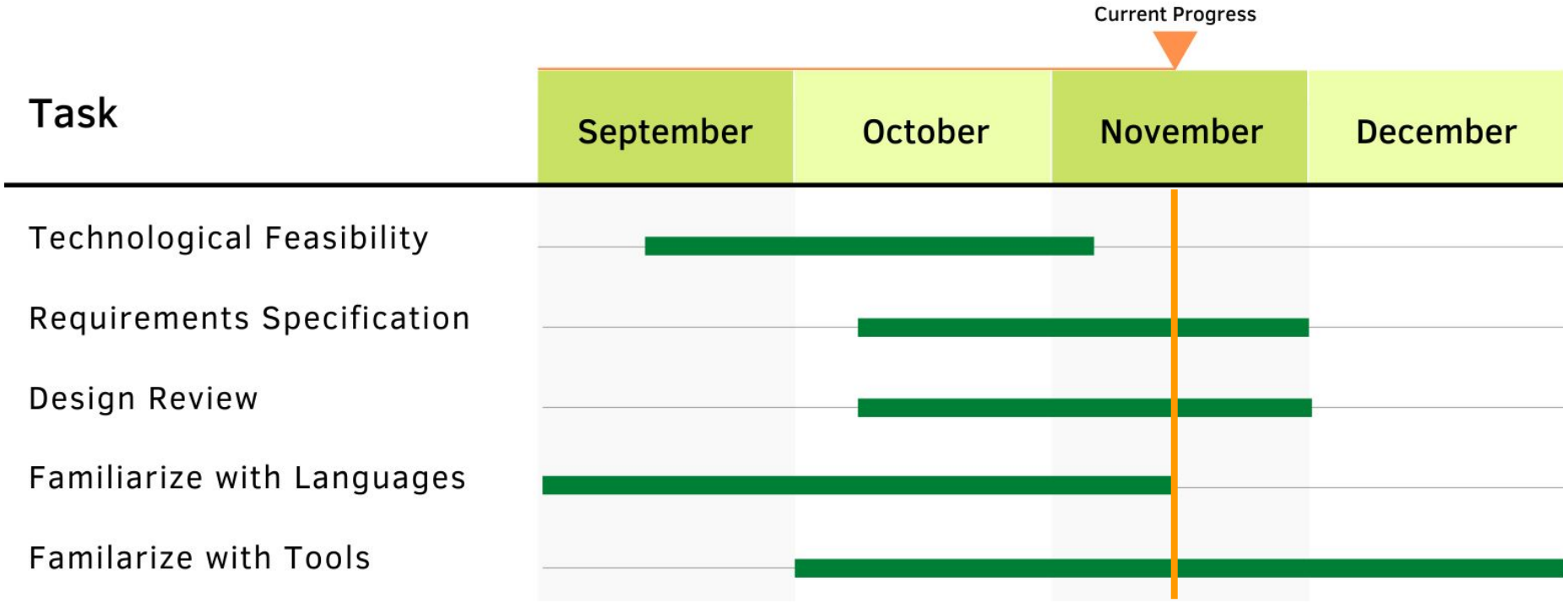
- Must be able to handle transfer of large files.
- Must be able to keep the connection open to wait for computations.
- Must be capable of running on many operating systems.



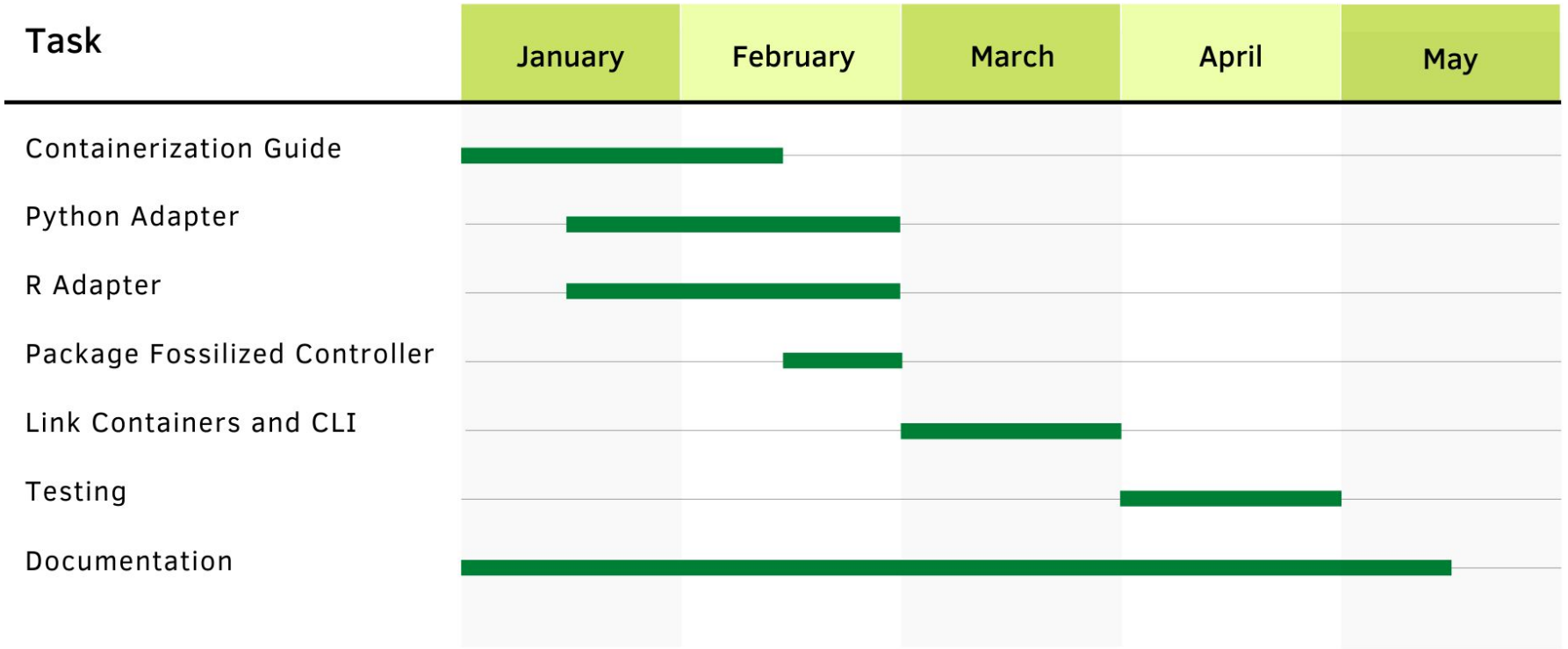
Risks

Risk	Likelihood	Effect	Solution
Poor Maintainability	**Low**	**High**	**Following proper coding standards and practices**
Corrupted Files	**Low**	**High**	**Good error checking and using a reliable library**
User Installation Issues	**Moderate**	**Moderate**	**Adding references to good troubleshooting resources**
Poor handling of large amount of files	**Low**	**Low**	**Using a reliable library**

Schedule


Current Progress
Task
September
October
November
December
Technological Feasibility
Requirements Specification
Design Review
Familiarize with Languages
Familarize with Tools

Schedule


Task
January
February
March
April
May
Containerization Guide
Python Adapter
R Adapter
Package Fossilized Controller
Link Containers and CLI
Testing
Documentation

Conclusion

- Paleoclimatology and model sharing
- Containers as solution
- Key Requirements
- Risks and Schedule
- Next:
  - Reviewing requirements with our sponsor
  - Continuing to build a prototype

Thank You

Questions?