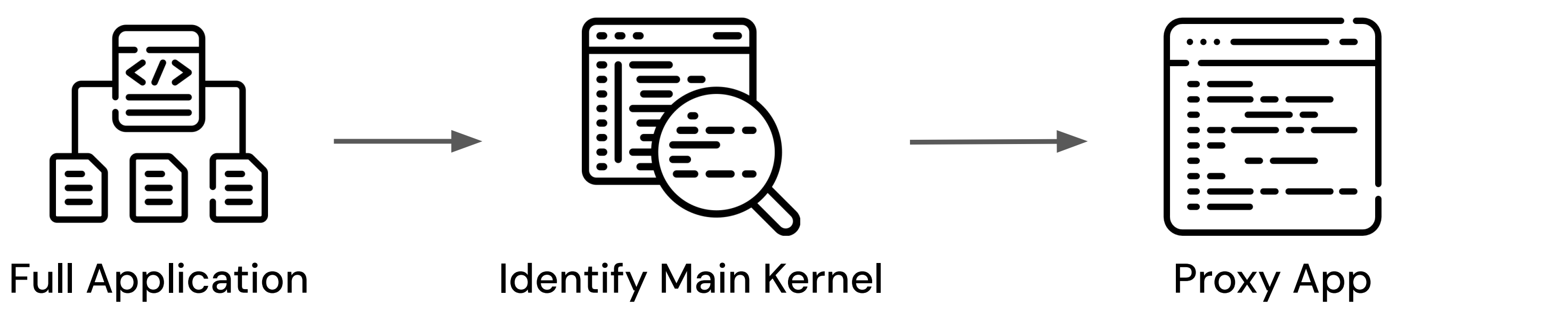
Characterizing Large Applications Through Proxy Benchmarks: a Pangenomic Case Study

Jessica Imlau Dagostini, Scott Beamer, Tyler Sorensen
Computer Science and Engineering Department
University of California, Santa Cruz

Introduction

- Large scientific applications are difficult to benchmark
- Involve complex inputs and many dependencies
- Can be inflexible to port to new hardware
- As a consequence, can be hard to be improved

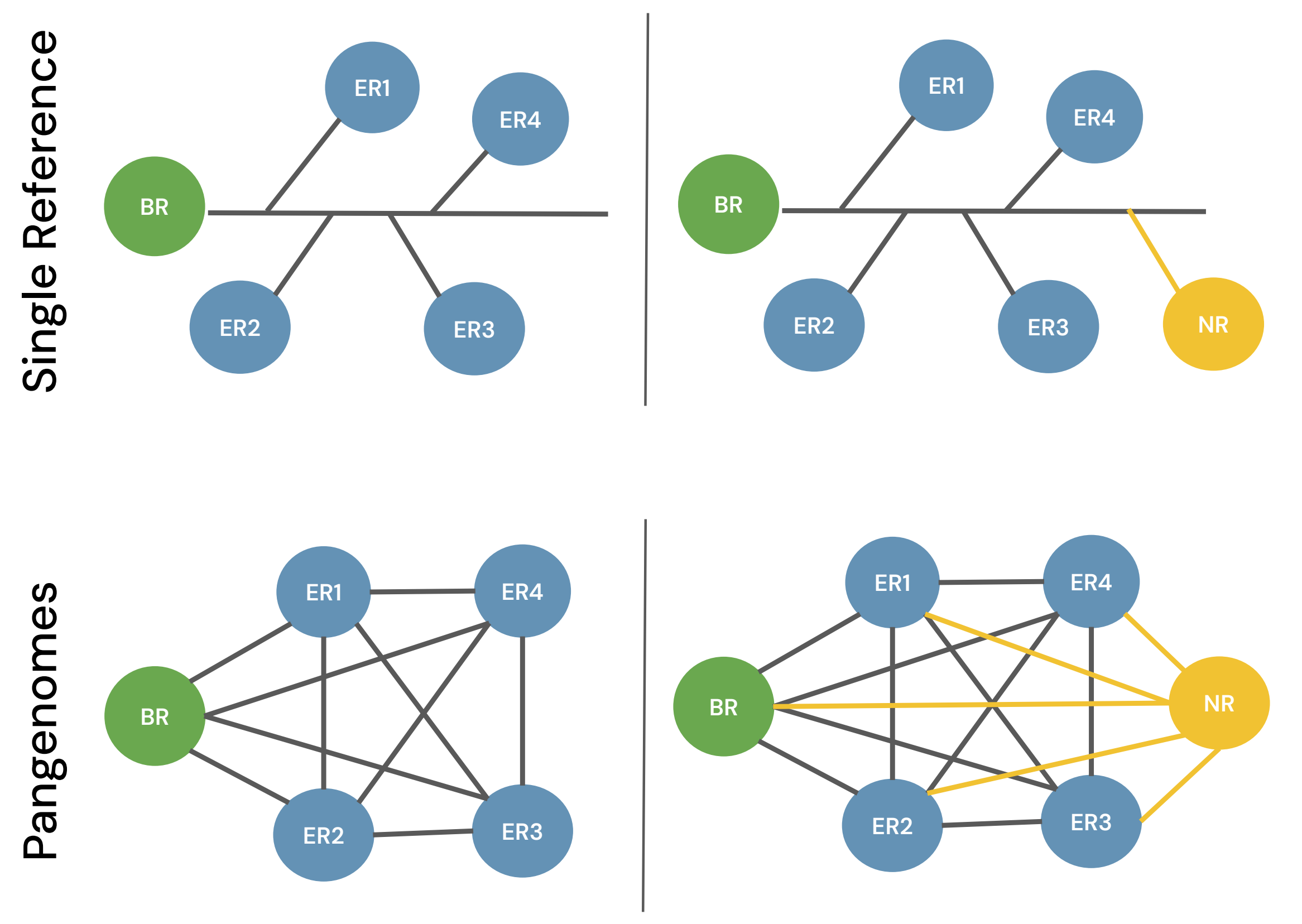
Proxy Apps can be a **lightweight** representation of real workloads that **ease performance** and **portability exploration**



In this work, we take the first steps to create a proxy application from an emerging and complex pangenomic application that maps short-read genomes to a pangenome graph reference.

Pangenomes

- Collection of common and unique genomes that are present in a given species [1]
- Composed of sequences from different individuals
- A more complete representation of a species' genome
- Complex DNA graph-based structure
  - Complexity to map sequences to this reference

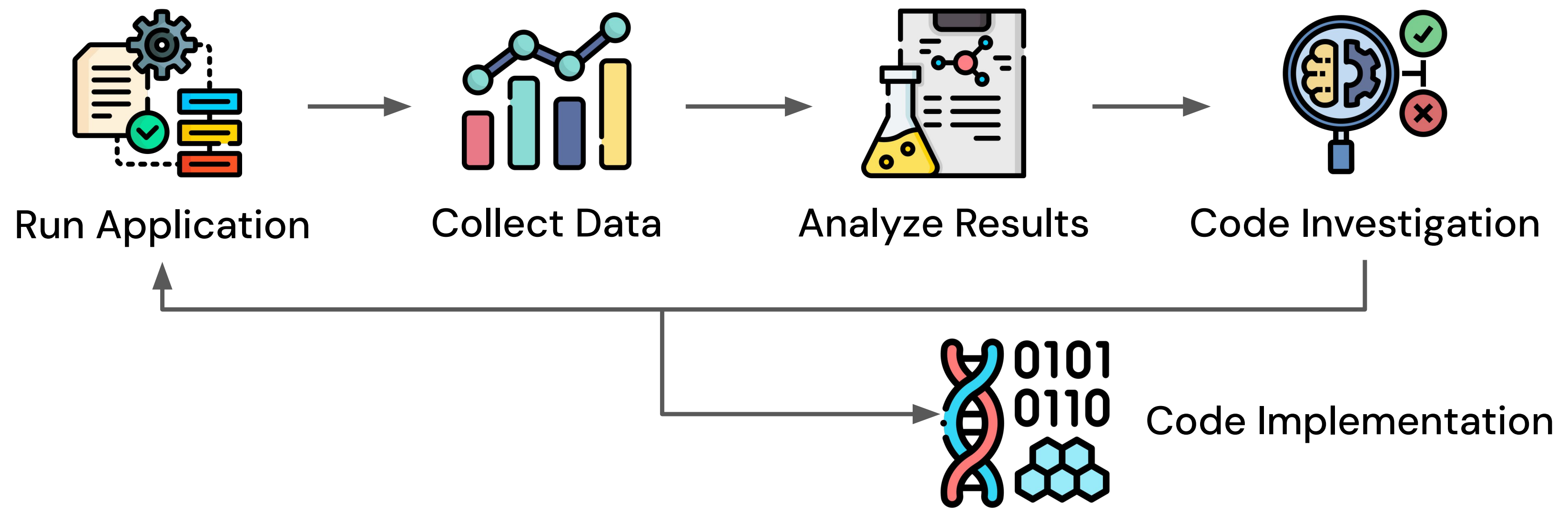


Single Genomes can be augmented with variations but do not contain all possible cross-references

Pangenomes allow to relate all genomes or sequences directly to each other

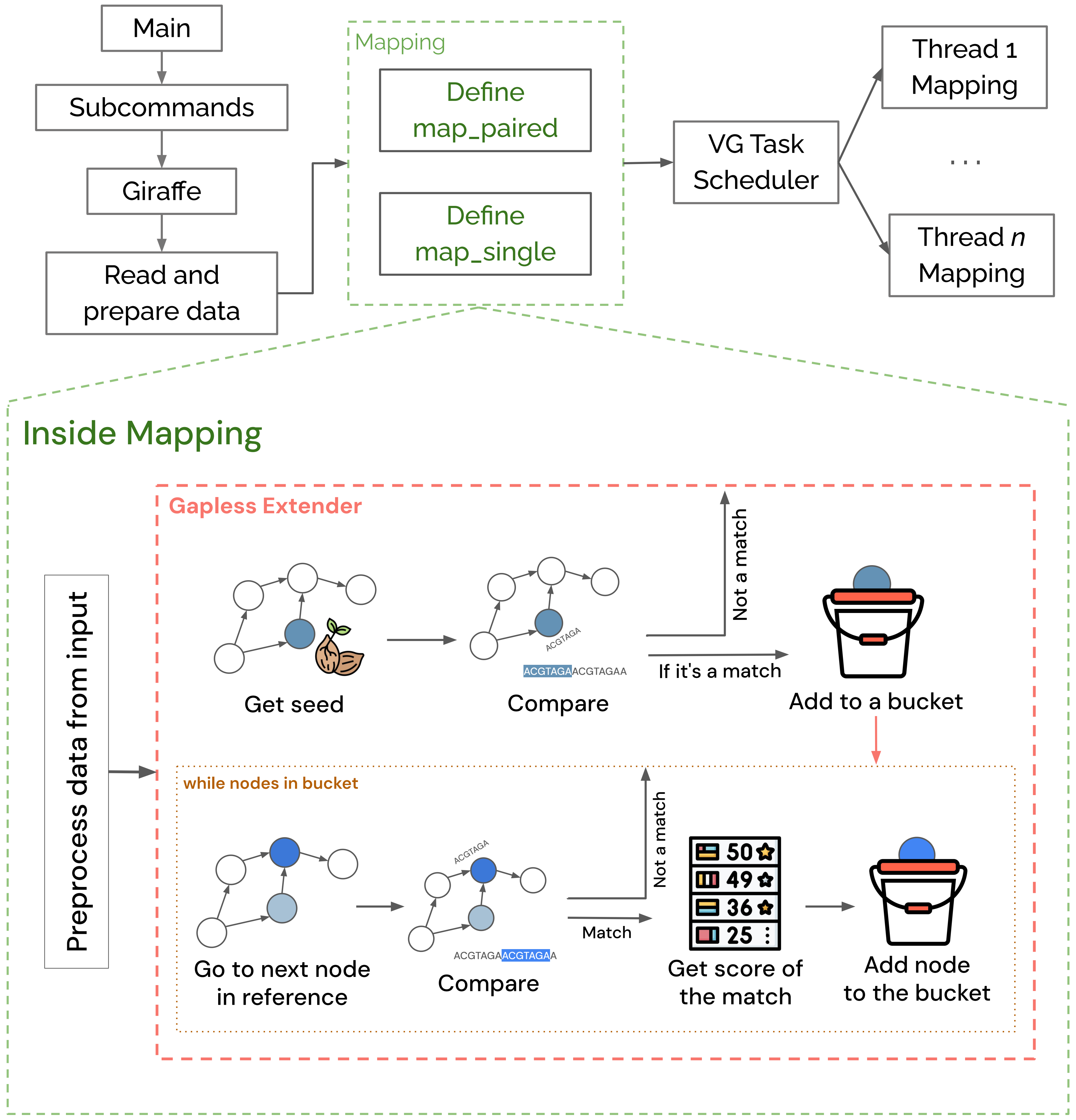
Source: Practical Graphical Pangenomics [https://pangenome.github.io/]

Proxy Generation Methodology



Pangenome Mapping Tool

We base our proxy app efforts on the *Giraffe* pangenome mapping tool [2]. Giraffe is one of the tools present in the *VG Toolkit* [3], which is a collection of tools focused on the creation and mapping of genome references as Variation Graphs



VG x Proxy

VG

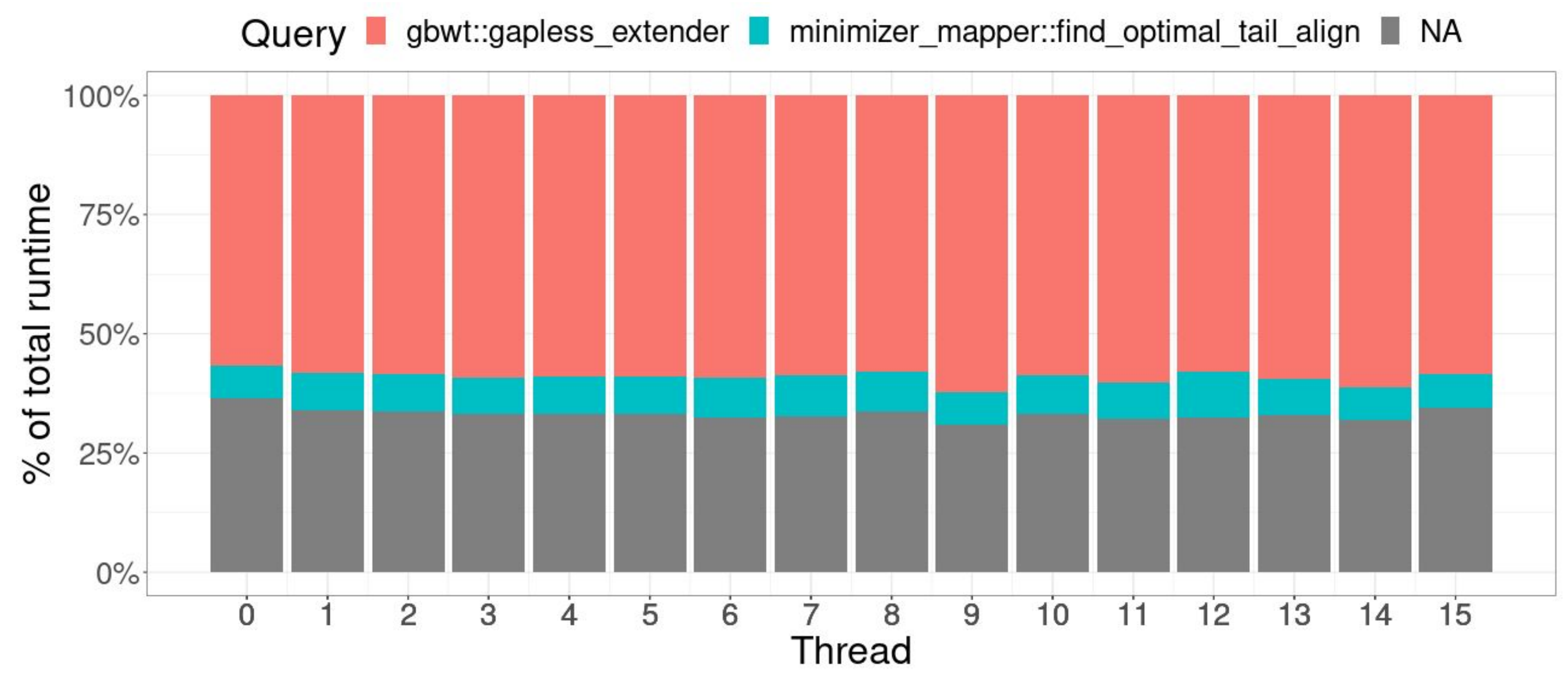
- ~50k lines of code
- ~350 source-files
- ~50 library dependencies
- Complex Makefile to compile
- More than 60s to compile
- Hard to profile

Proxy

- ~1k lines of code
- 2 source-files
- 3 library dependencies
- Simple Makefile to compile
- Around 10s to compile
- Easy to profile and play with different strategies

Profiling Results

To identify the most important mapping code within Giraffe, we profile the two functions that access the pangenome graph: the *gapless_extension* function and the *find_optimal_tail_alignment* function. *gapless_extender* represents more than 60% of the application's runtime in all threads.



[1] Abondio P, Cilli E, Luiselli D. Human Pangenomics: Promises and Challenges of a Distributed Genomic Reference. Life (Basel). 2023 Jun 9;13(6):1360. doi: 10.3390/life13061360. PMID: 37374141; PMCID: PMC10304804.
[2] Jouni Sirén et al. ,Pangenomics enables genotyping of known structural variants in 5202 diverse genomes.Science374,abg8871(2021).DOI:10.1126/science.abg8871
[3] Garrison, E., Sirén, J., Novak, A. et al. Variation graph toolkit improves read mapping by representing genetic variation in the reference. Nat Biotechnol 36, 875–879 (2018). https://doi.org/10.1038/nbt.4227
[4] Cook, Jeanine, Finkel, Hal, Junghams, Christoph, McCorquodale, Peter, Pavel, Robert, & Richards, David F. Proxy App Prospectus for ECP Application Development Projects. United States. https://doi.org/10.2172/1477829

We sincerely thank Dr. Joseph B Manzano (Pacific Northwest National Laboratory) for assisting with most of the profiling and code investigation done in this work.

Scan this QR code to access the PDF version of this poster