# Genomics Without Borders

Nathan Hammond, PhD; Isaac Liao, PhD; Ziliang Qian, PhD; Somalee Datta, PhD
Stanford Center for Genomics and Personalized Medicine

STANFORD SCHOOL OF MEDICINE

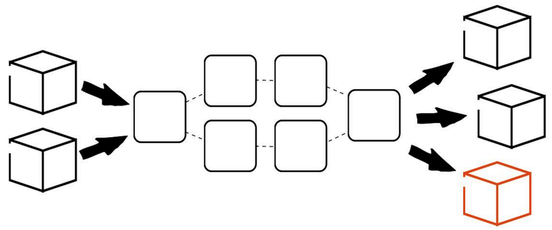
## Genomics means big data

Genomics is moving from the university to the clinic, and from local hardware to the cloud. Datasets now routinely reach the petabyte scale. With much of the bioinformatics community still surviving on bash scripts and Makefiles, analysts are looking for new tools that support the scale and requirements of genomics analysis.

| Project | Data volume/ year | PHI | On-premise | Cloud | 3rd-party analysis tools |
|---|---|---|---|---|---|
| VA Million Veteran Program | ~1 PB | ✔ | ✔ | ✔ | ✔ |
| iPoP | ~1 PB | | ✔ | ✔ | ✔ |
| Stanford Clinical Genomics Service | ~1 PB | ✔ | ✔ | ✔ | ✔ |
| Stanford Sequencing Center | ~100 TB | | ✔ | ✔ | ✔ |

## Genomics workflows

Data is just data. So how is Genomics different from other Big Data domains?



- Analyses are a concatenation of serial and parallel steps
- Reliance on command line tools, often academic, almost always using file IO
- Embarrassingly parallel workflows
- Each sample produces many 100-GB flat files with PHI
- Frequently update and rerun analyses as algorithms improve

## Privacy

Genomes can sometimes be traced back to an individual using publicly available resources. Genomic data therefore cannot easily be de-identified, and should be handled securely. Many organizations consider genomic data to be Personal Health Information subject to HIPAA regulation.

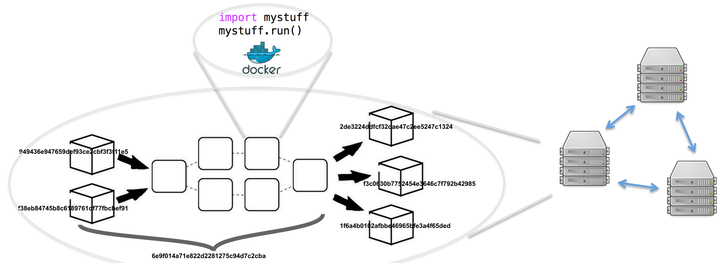
Solutions for secure storage and processing of very large datasets are not yet mature.

## XPPF: eXtremely Portable Pipeline Framework

XPPF is designed to address the following:

- Centralized framework handles common functions
  - encryption
  - data transfers
  - logging
  - dependency management
  - failure recovery
- Results lookup to avoid rerunning redundant analysis steps
- Portability of analyses and results across different clouds, clusters, and server instances
- Repeatability and verifiability of analysis
- File provenance tracking

## Key features of XPPF




**Docker**
By running analysis steps in Docker containers with all the needed software dependencies, pipelines are portable across clusters, independent of environment.

**Documents, not code**
Analysis pipelines are expressed as a JSON document.

**Content-based hashing**
Each input file, result, and analysis specification is identified by a hash of its contents. This enables several useful features

- Objects transferred between servers with independent databases keep the same primary key and retain the same relationships, if related objects are available.
- Duplicate analysis requests are easily streamlined to avoid duplicate processing. Multiple requests for the same analysis will link to the same analysis object. There is no need to rerun analysis for later requests. This is especially useful for common tasks like indexing a reference genome, or for running a published analysis pipeline on a public dataset.
- Traceability and repeatability are enhanced by the fact that all inputs can be verified by their hash.
- Unknown objects can be looked up by their hash. This is especially useful for finding how a particular result file was generated.

## Pipelines that run anywhere

Writing pipelines as JSON documents, not as code, enables clean separation of the analysis description from the workflow implementation. Many organizations have coalesced around the idea of creating a pipeline description language (e.g. participants in the Common Workflow Language initiative).

Vendors and developers of workflow engines who use JSON or XML as a pipeline expression language and use Docker for application portability will be able to translate pipelines from one language to another, making analysis pipelines portable across workflow engines.

## Discussion

We are on the cusp of a major change in how bioinformatics pipelines are written, where they run, and how data can be shared.

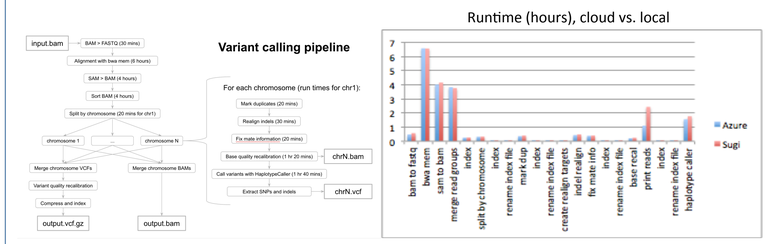
These are some of the factors driving the change:

- The low cost and high data volume of Next Generation Sequencing has caused an explosion in genomics and increased the need for both research and clinical analysis.
- Docker, although still immature, has made it trivial to port applications from one system to another, and to run each step of a pipeline in a unique environment.
- Falling cost of cloud services, increasing level of comfort with working in the cloud, and the impracticality of shipping petabytes of data between isolated HPC clusters is driving genomics analysis to the cloud.

XPPF is one of several projects helping to change this space, but common features are emerging in many solutions: Docker, JSON-based analysis descriptions, and (to a lesser extent) identifying objects with a content-based hash to enable provenance tracking and eliminate redundant analysis runs.

## XPPF in action

XPPF is still under development. As a proof of concept, we recently ran a variant calling pipeline modeled on GATK best practices. The same analysis was executed on Microsoft Azure and on a local cluster at Stanford without modification.




## Contact

Nathan Hammond
Stanford Center for Genomics and Personalized Medicine
nhammond@stanford.edu
https://github.com/StanfordBioinformatics/xppf