

Persona: A High-Performance Bioinformatics Framework

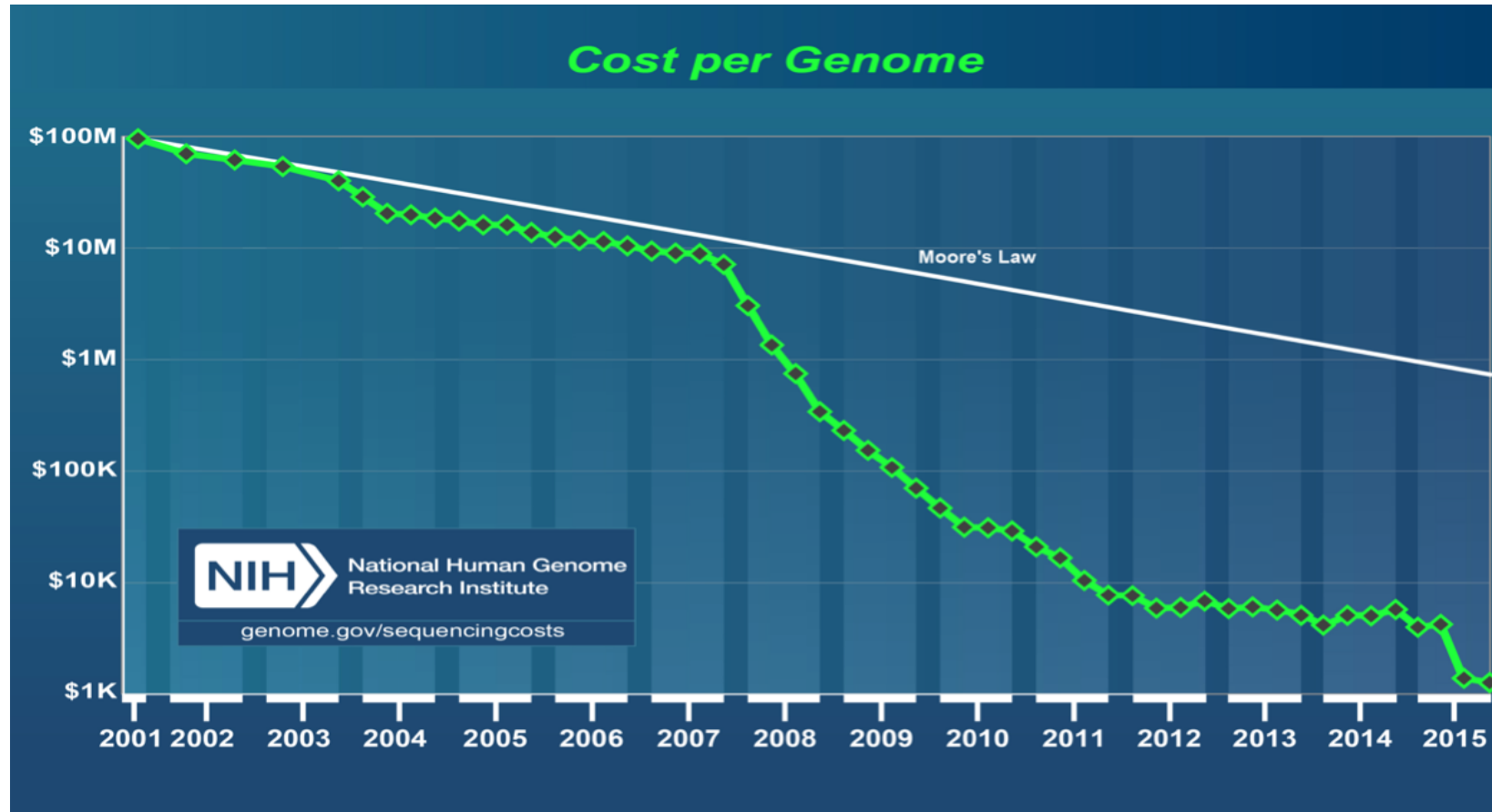
Stuart Byma¹, Sam Whitlock¹, Laura Flueratoru²,
Ethan Tseng³, Christos Kozyrakis⁴, Edouard Bugnion¹, James Larus¹

EPFL¹, U. Polytechnica of Bucharest², CMU³, Stanford⁴

Agenda

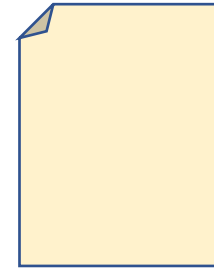
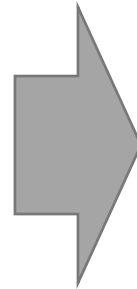
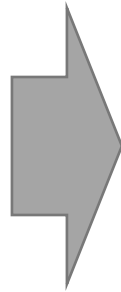
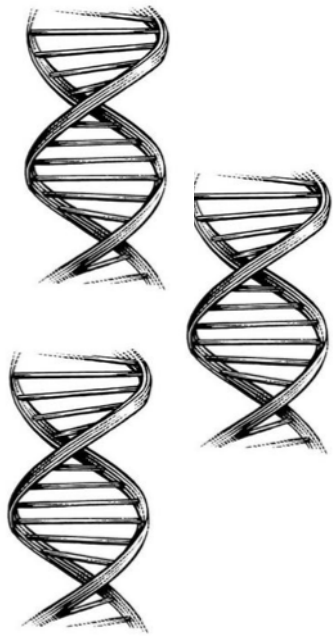
- **Motivation**
- Bioinformatics Data and Tools
- Persona
 - AGD
 - Dataflow Engine
- Performance Results

Sequencing cost

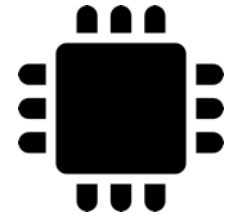


Not a wet lab problem anymore → IT / Systems problem

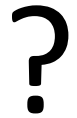
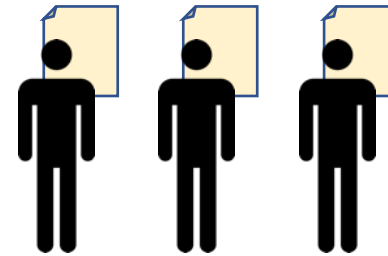
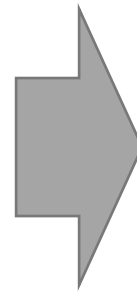
Implications



~300GB



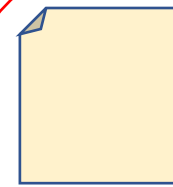
~hours



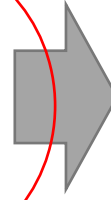
Need efficient systems that scale well

Agenda

- Motivation
- **Bioinformatics Data and Tools**
- Persona
 - AGD
 - Dataflow Engine
- Performance Results



~300GB



~hours

What kind of data?

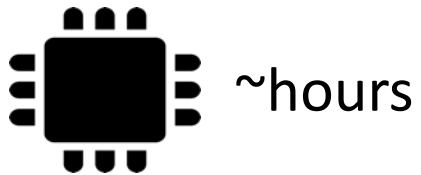
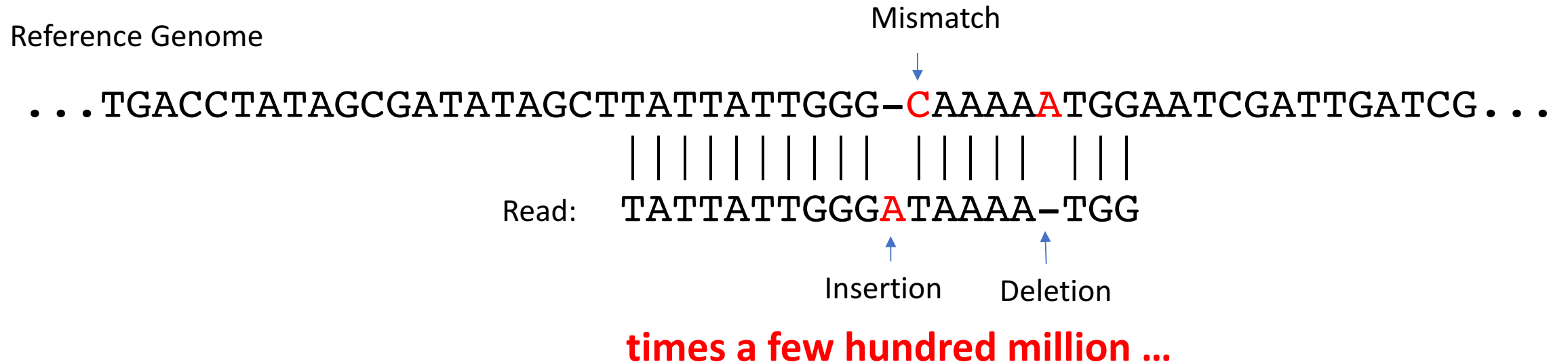
- Common sequencers produce *Reads*
 - Snippets of DNA → AACCGCTAGCGCGCTAGCTCGAGCTAGAA
 - 100-200 bases



```
@sequence name, metadata  
ACGTTTCGATCGCGCCAGGAGGCTAG  
+  
-+* ' ' ) ) **55CCF@>>>>>CCCCC
```

times a few hundred million ...

Alignment

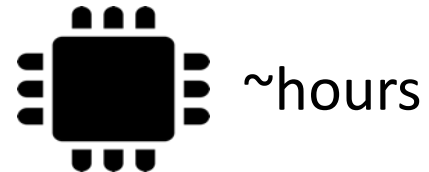


Aligned Reads

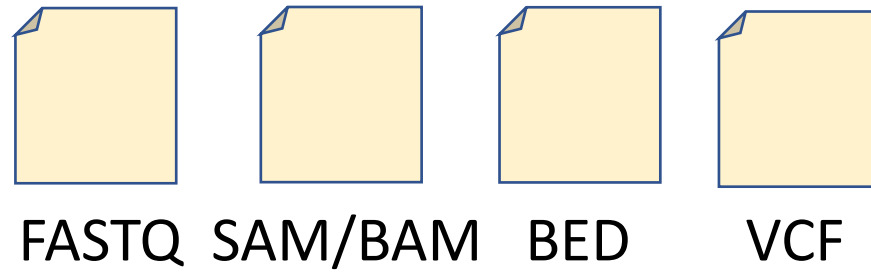
- Stored in SAM/BAM

```
read_name 16      chr12      85500011      70
18M      *      0      0
TTTACACACATTATCTC      CDDFAEEC>EDDFFBCDEED?FCC@
PL:Z:Illumina      PU:Z:pu LB:Z:lb SM:Z:sm
```

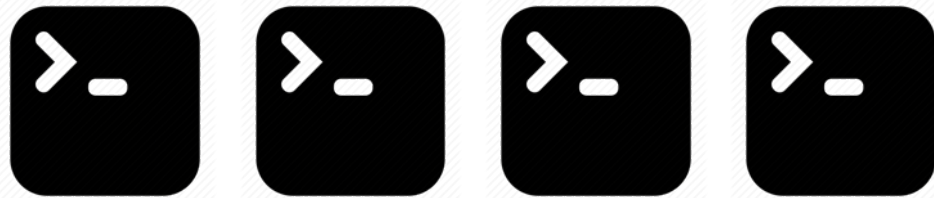
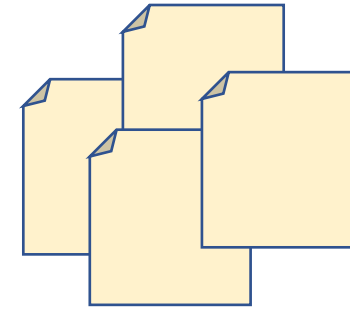
- Followed by
 - Duplicate marking
 - Sorting
 - Recalibrations, analysis (variant calling)



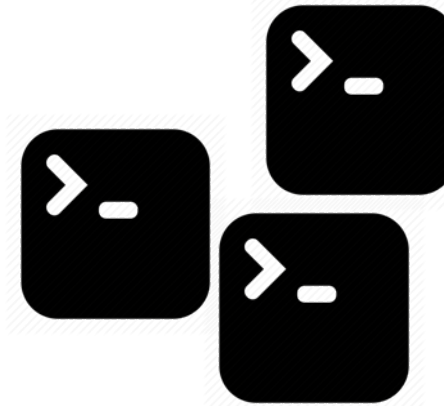
Data and Tool Issues



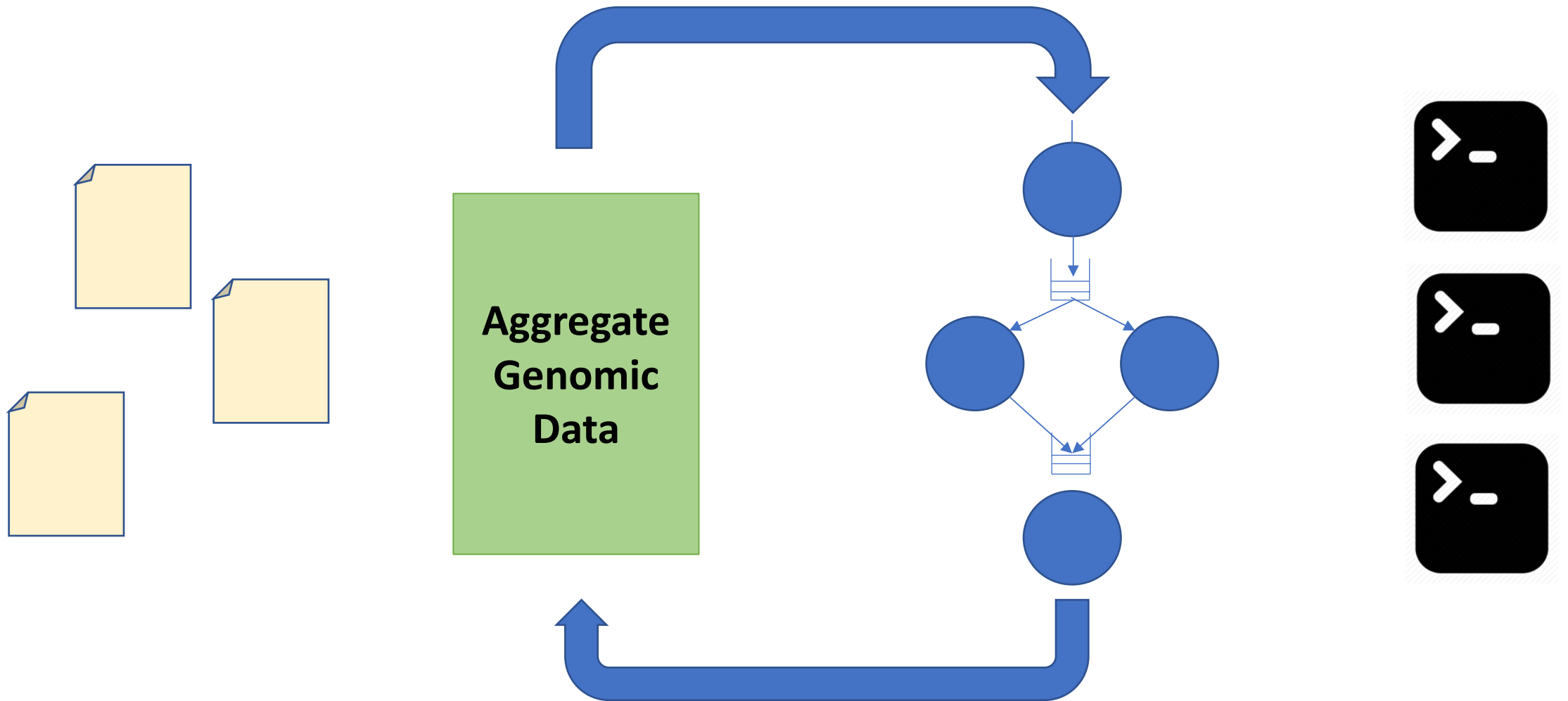
...



...



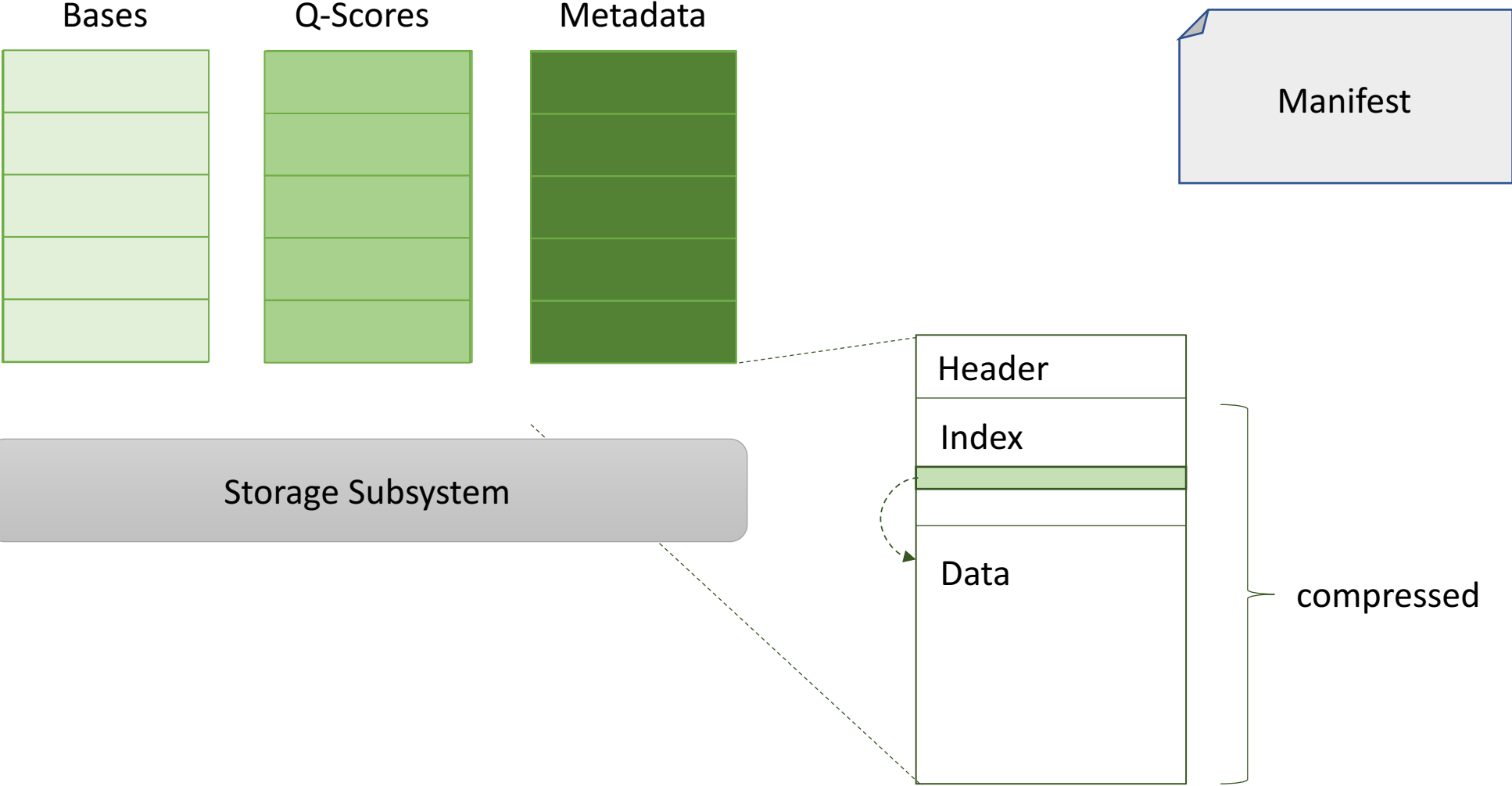
Persona – Bioinformatics, Unified



Agenda

- Motivation
- Bioinformatics Data and Tools
- Persona
 - **AGD**
 - Dataflow Engine
- Performance Results

Aggregate Genomic Data



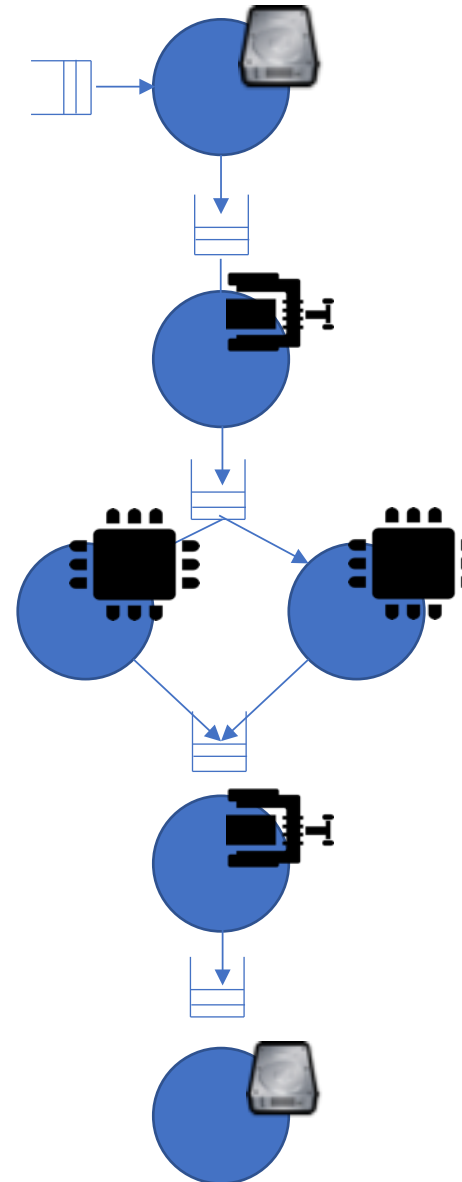
Agenda

- Motivation
- Bioinformatics Data and Tools
- Persona
 - AGD
 - **Dataflow Engine**
- Performance Results

Dataflow

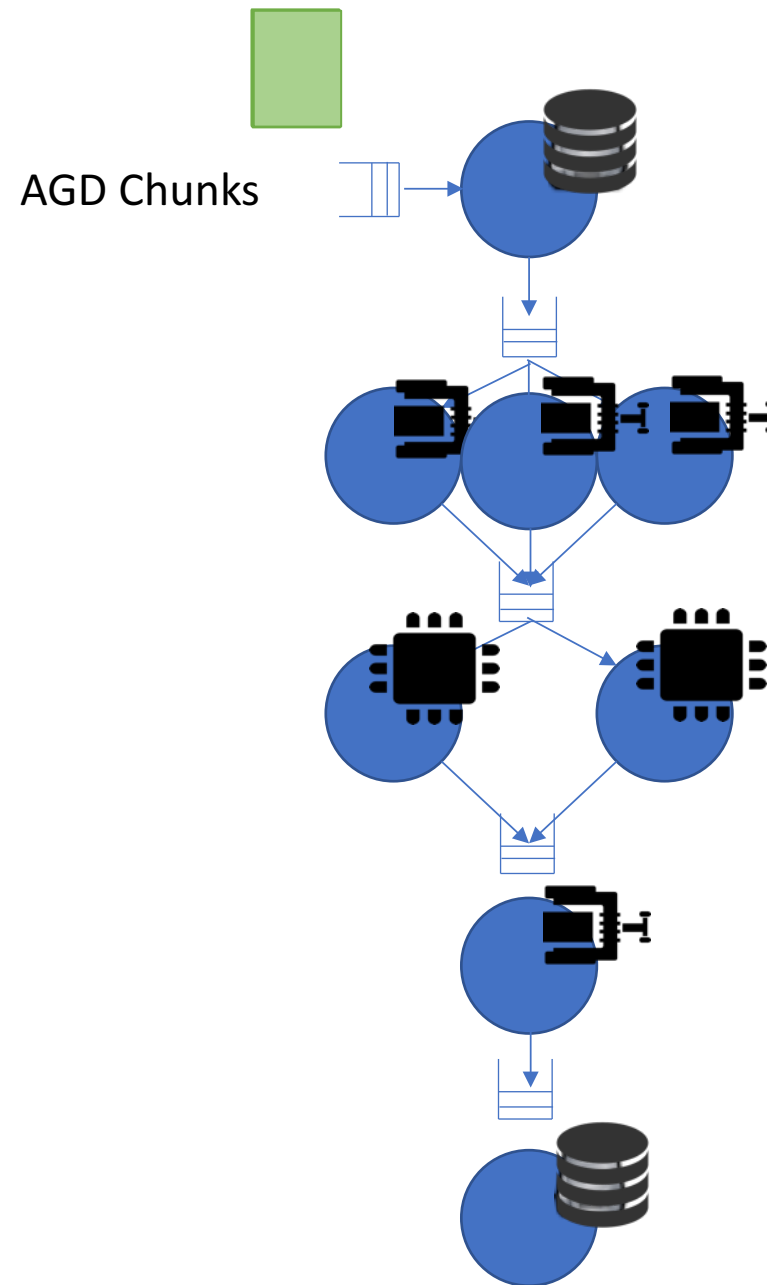
- Dataflow execution framework
 - Base on TensorFlow engine
 - But no machine learning
- Operators perform computation on AGD chunks

AGD Chunks



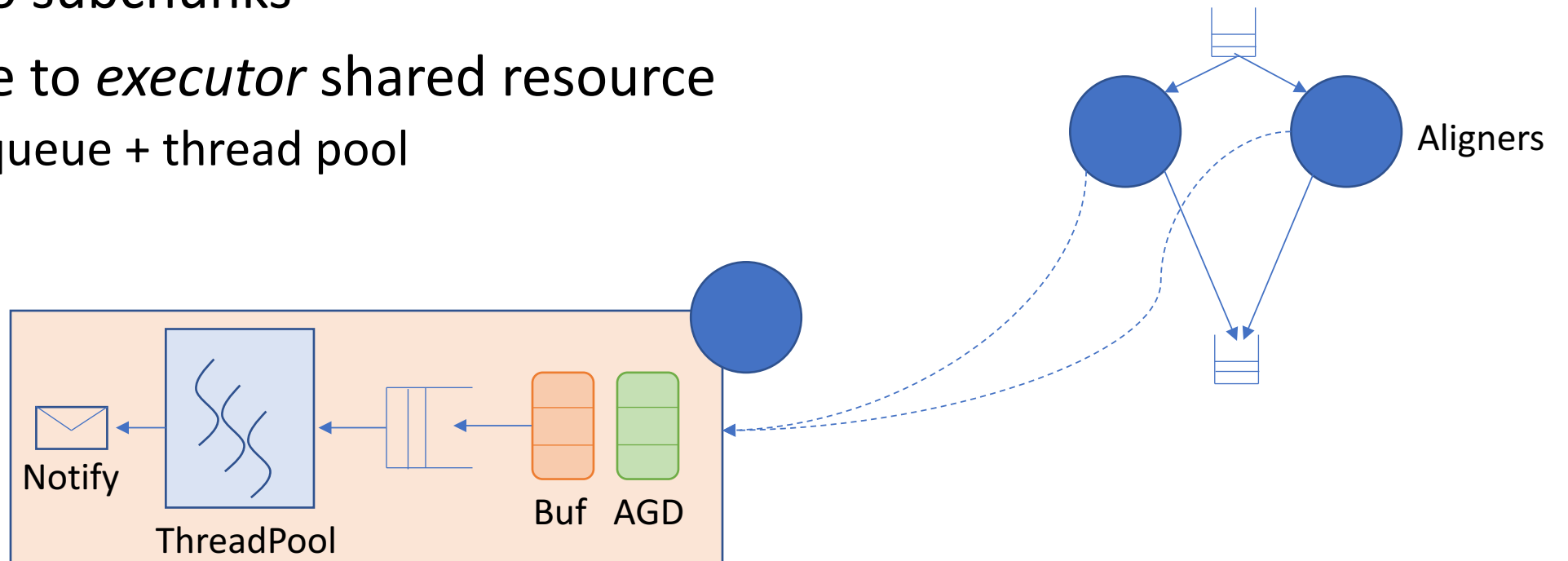
Dataflow

- Modularity
- Balance/tuning
- (bounded) Queueing

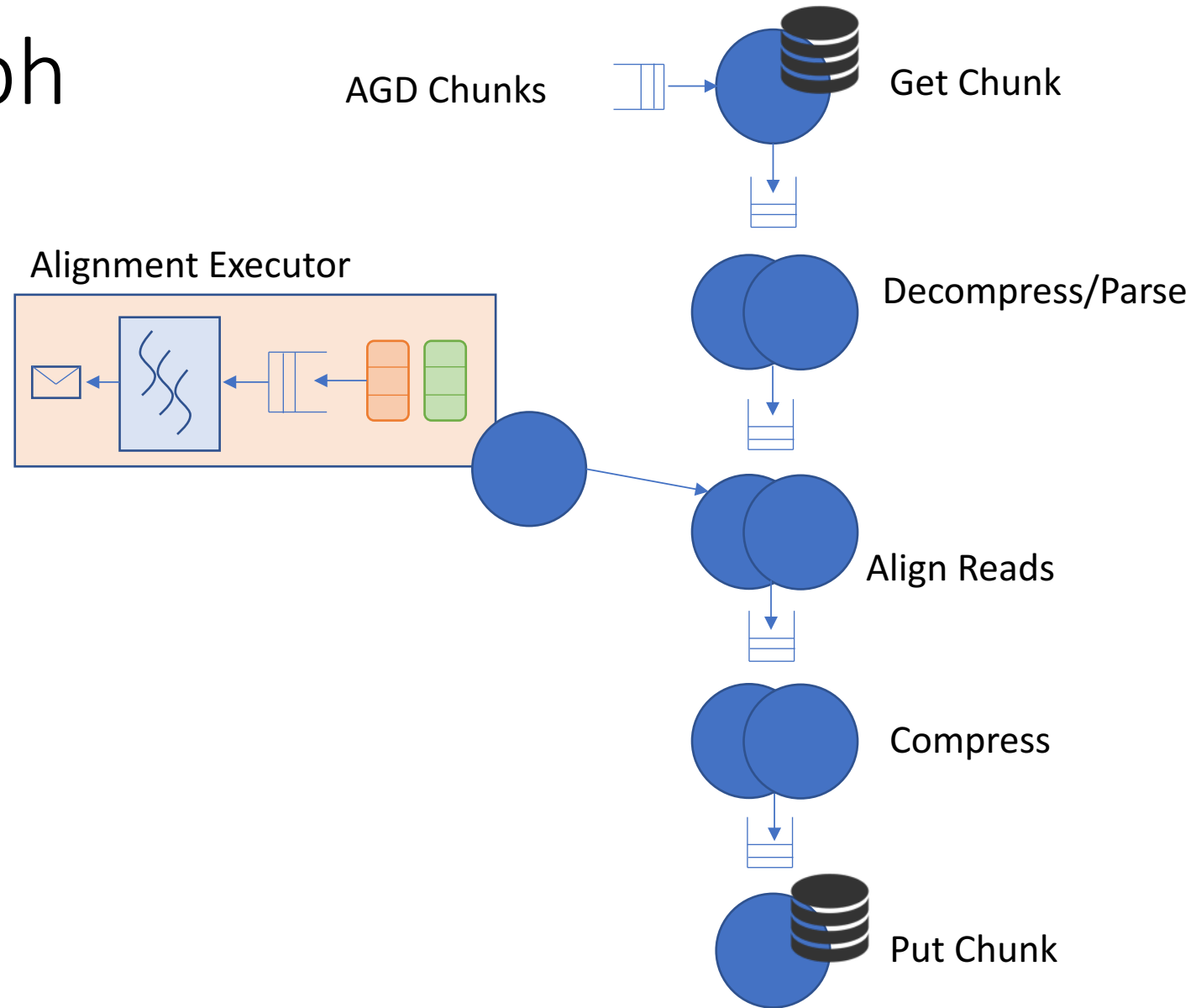


Fine-grained Threading

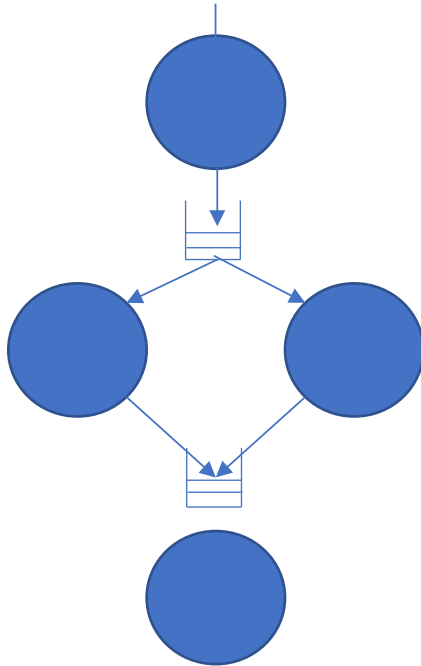
- AGD chunks optimized for storage
 - Too coarse for some tasks
- Split into subchunks
- Delegate to *executor* shared resource
 - Task queue + thread pool



Aligner Graph



Graph Construction



```
c = persona.read_chunk(path)
```

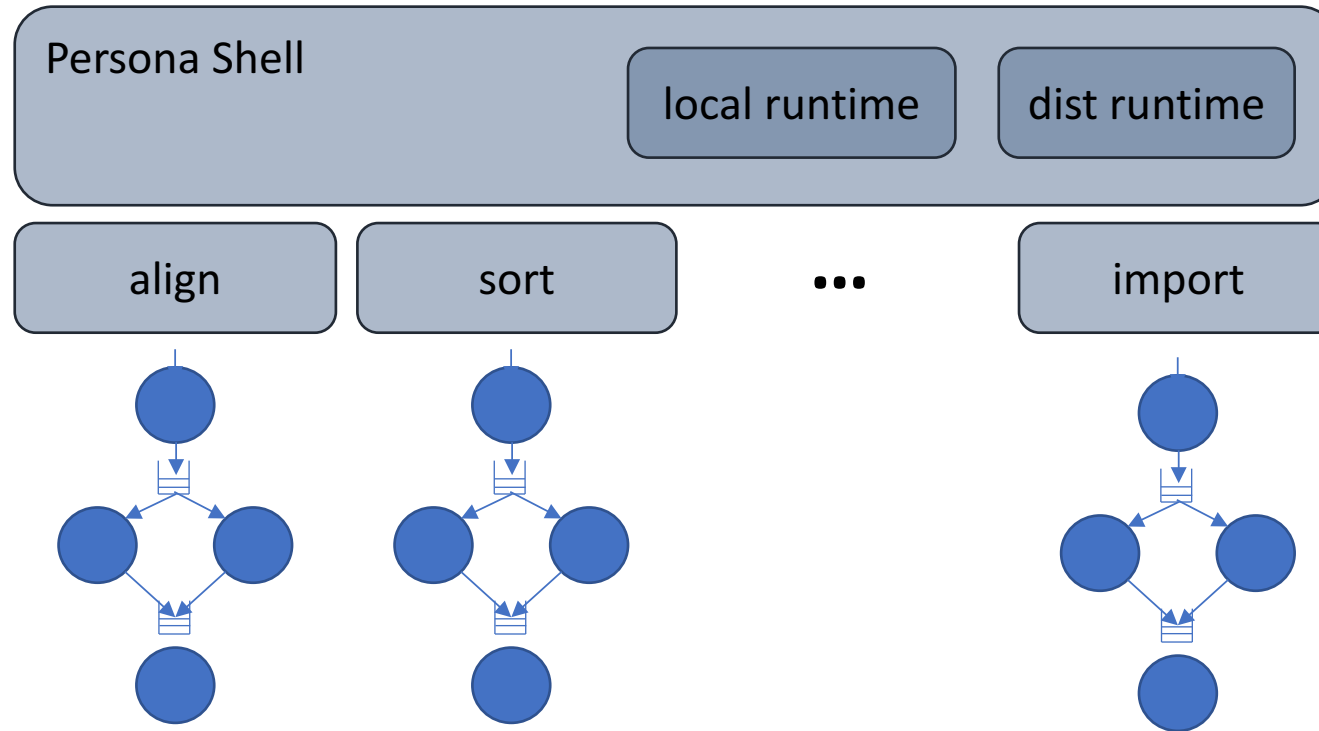
```
d = persona.decompress(c)
```

```
o = persona.align(d)
```

```
sess = tf.Session()
```

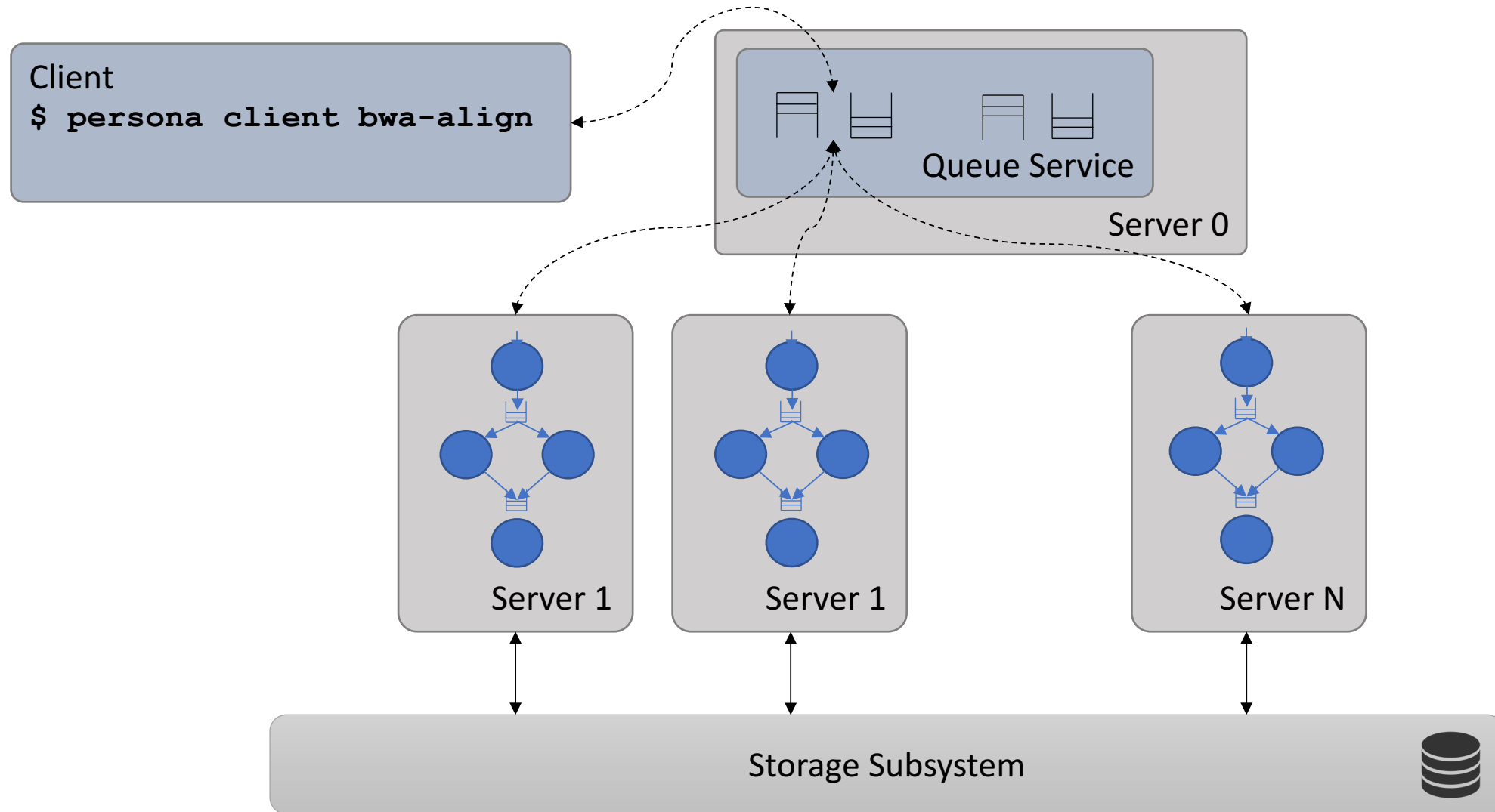
```
result = sess.run([o])
```

Persona Shell



```
$ persona align local -i hg19 data/my_agd.json  
$ persona sort local data/my_agd.json
```

Distributed Computation



Current Features

- Import data from FASTQ/BAM/SRA, export to BAM
- Sequence alignment with BWA-MEM, SNAP
- Dataset sorting
- Duplicate marking
- Dataset statistics (samtools flagstat)
- Read coverage (depth)

Agenda

- Motivation
- Bioinformatics Data and Tools
- Persona
 - AGD
 - Dataflow Engine
- **Performance Results**

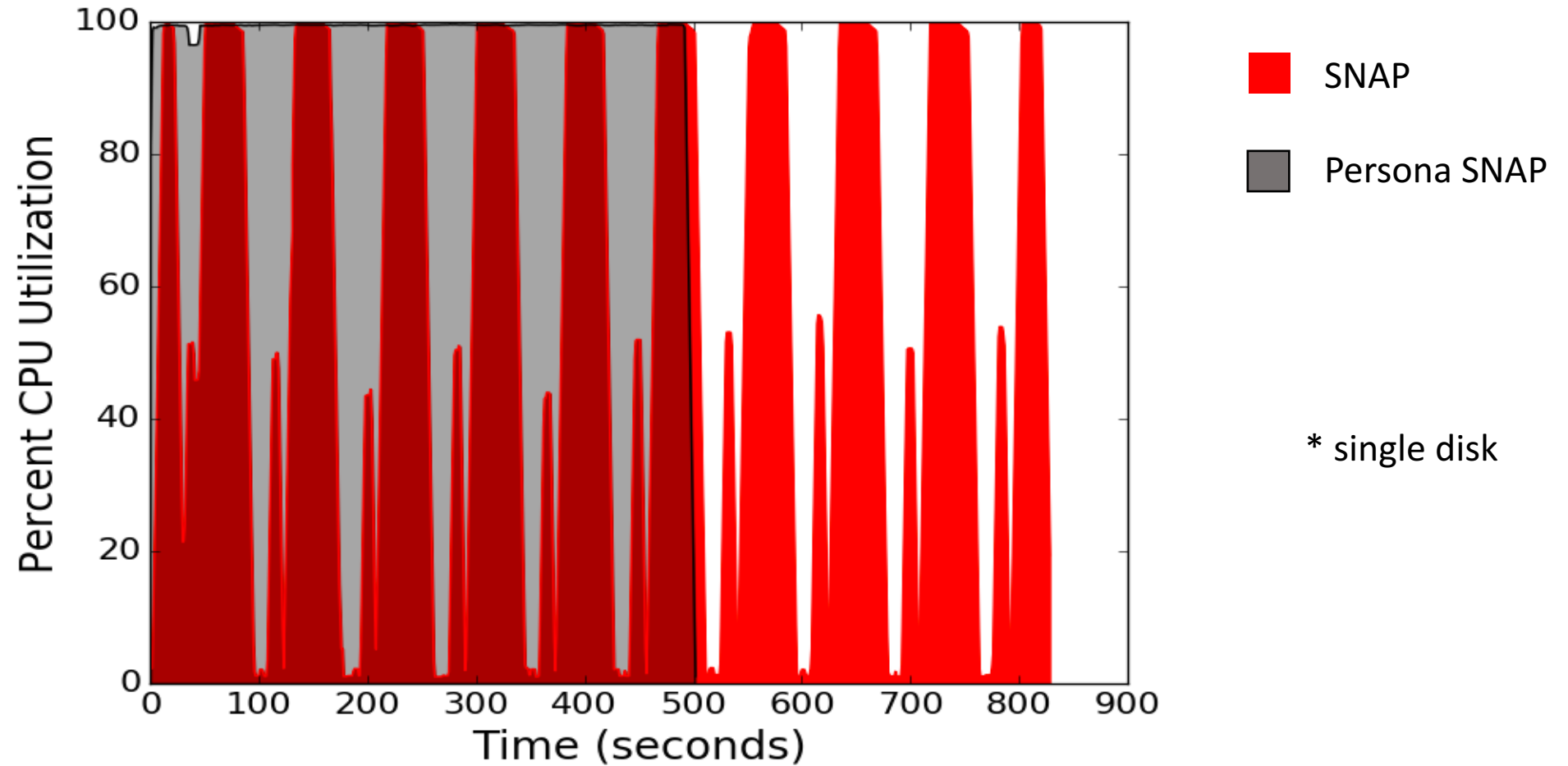
Evaluation -- Setup

- Focused on sequence alignment using SNAP
- Throughput in bases aligned per second
- Data
 - 223 million 101 base reads (~16GB)
 - AGD chunks of 100K records
- Hardware
 - 32X Ubuntu 16.04, 2x12 Xeon E5-2680v3 @ 2.5GHz
 - Data on 6-disk RAID0 and single spindle drive
 - 7 server Ceph object store for distributed execution

Evaluation -- Questions

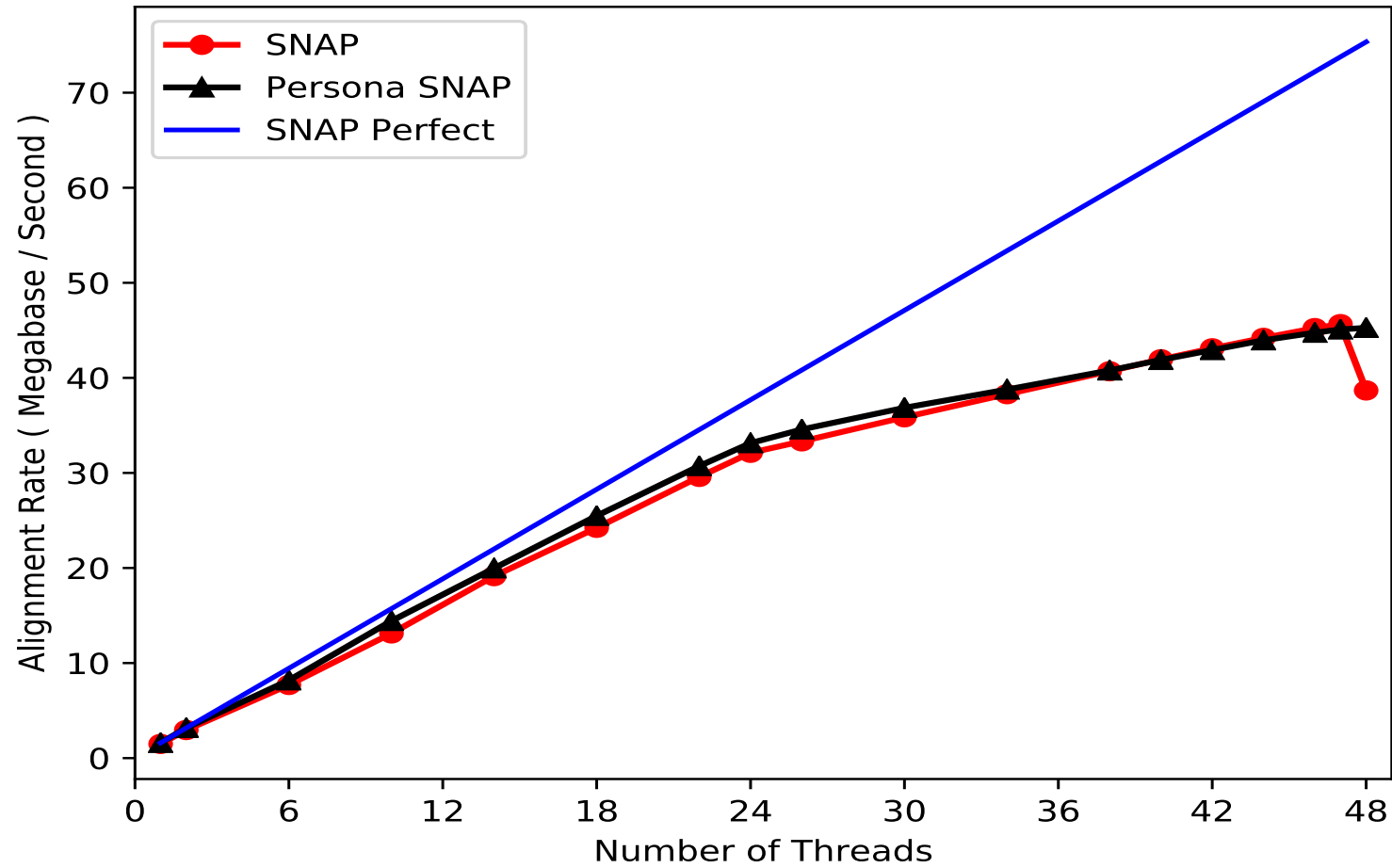
- What are the bandwidth-saving effects of AGD?
- What is the overhead of the Persona framework?
- How well do Persona and AGD scale?

Performance – AGD



Significantly less I/O → more efficient use of HW, BW

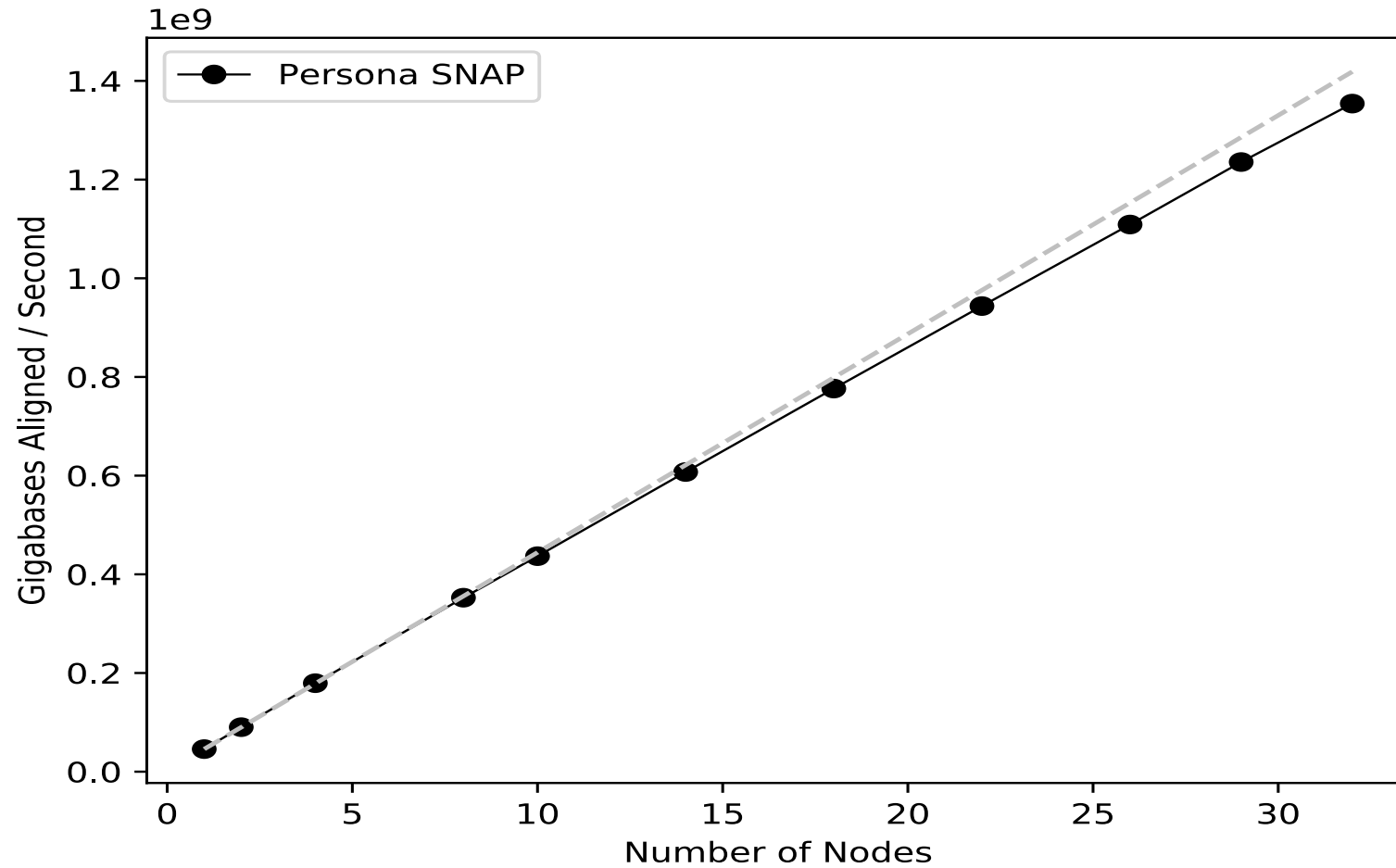
Persona Overhead



* RAID-0

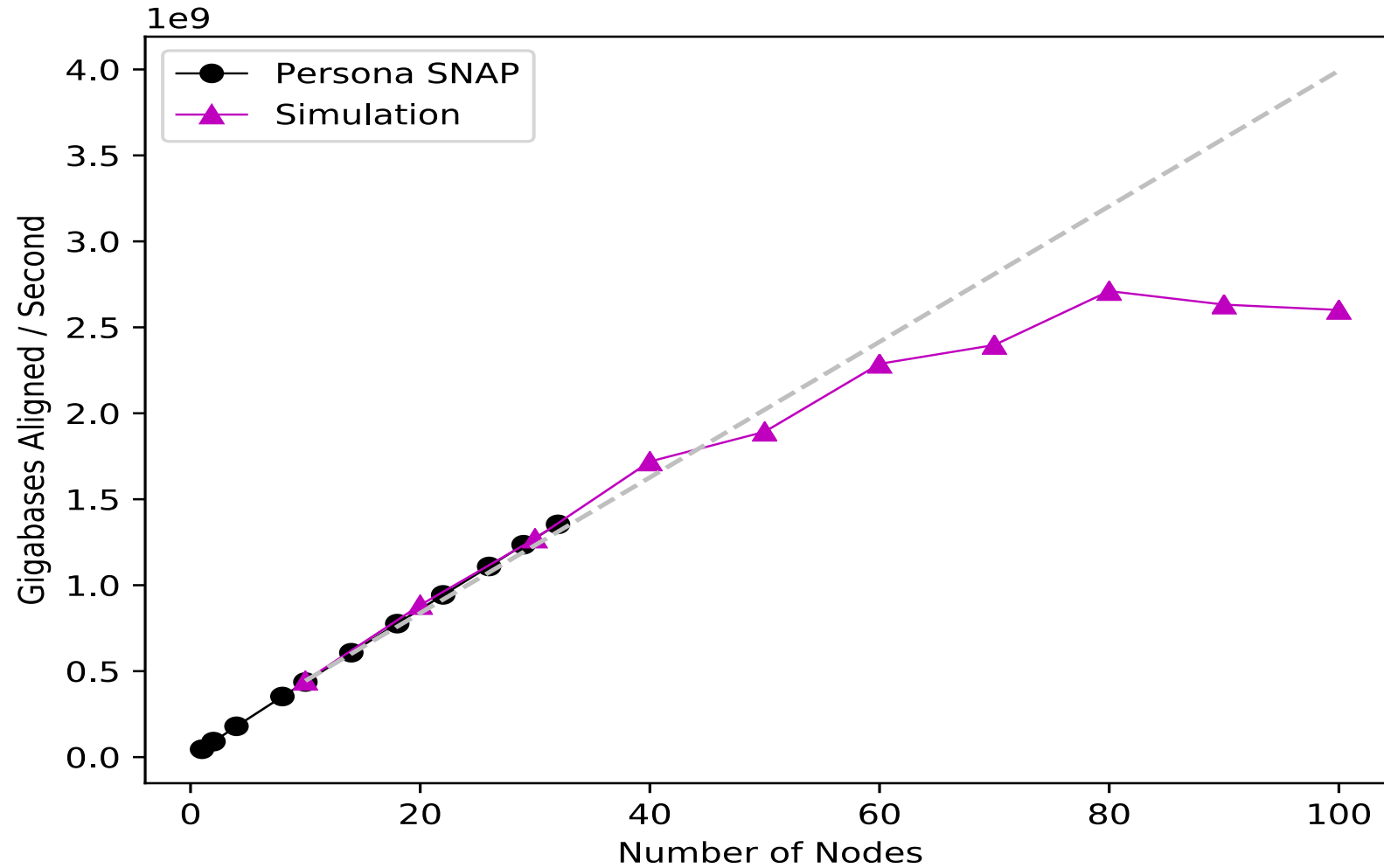
Negligible overhead!

Scaling

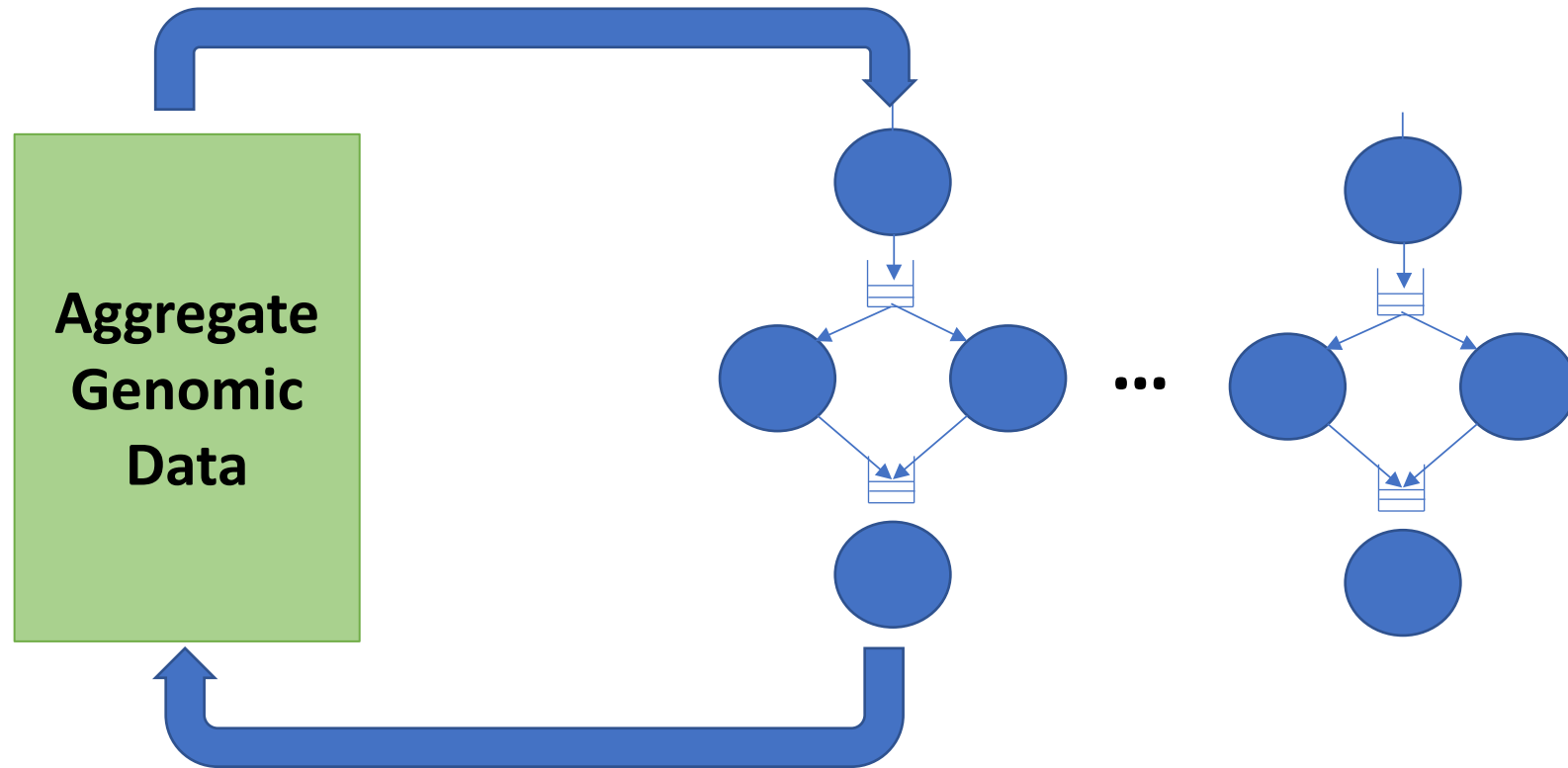


Full dataset aligned in ~17 seconds

Scaling Limits





Persona – Scalable Bioinformatics



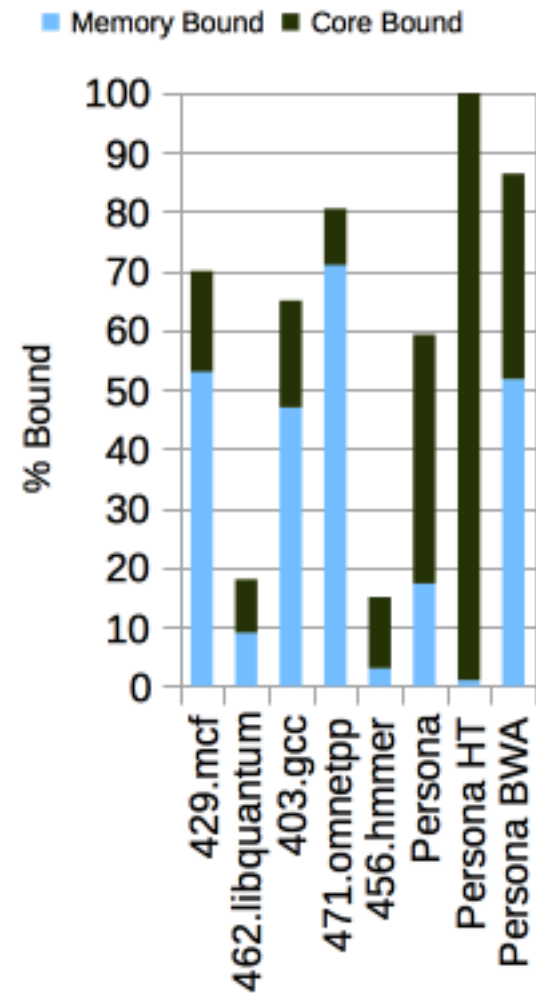
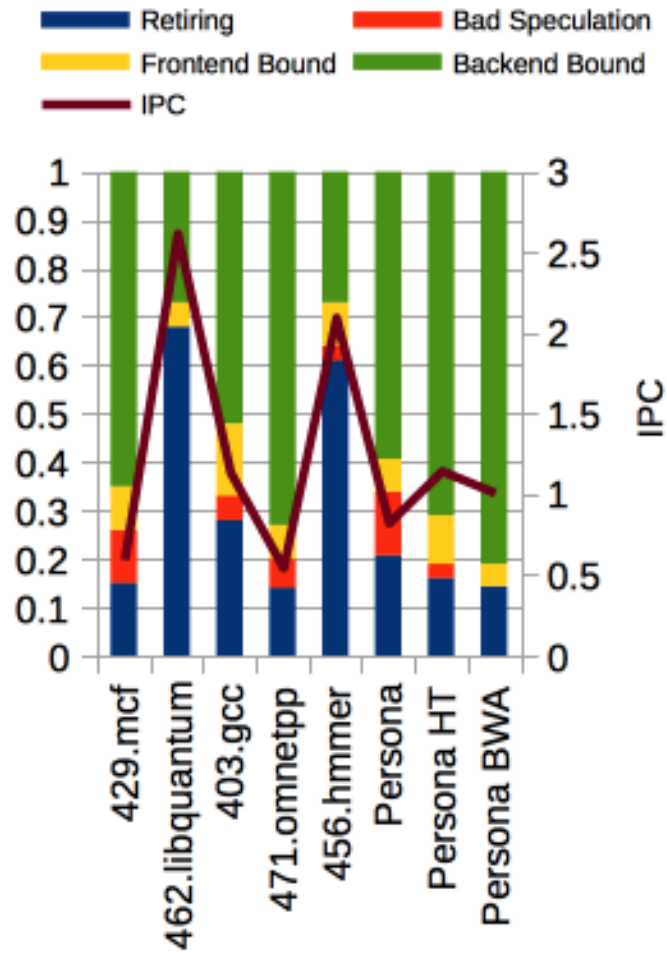
<https://github.com/epfl-vlsc/persona>

backup

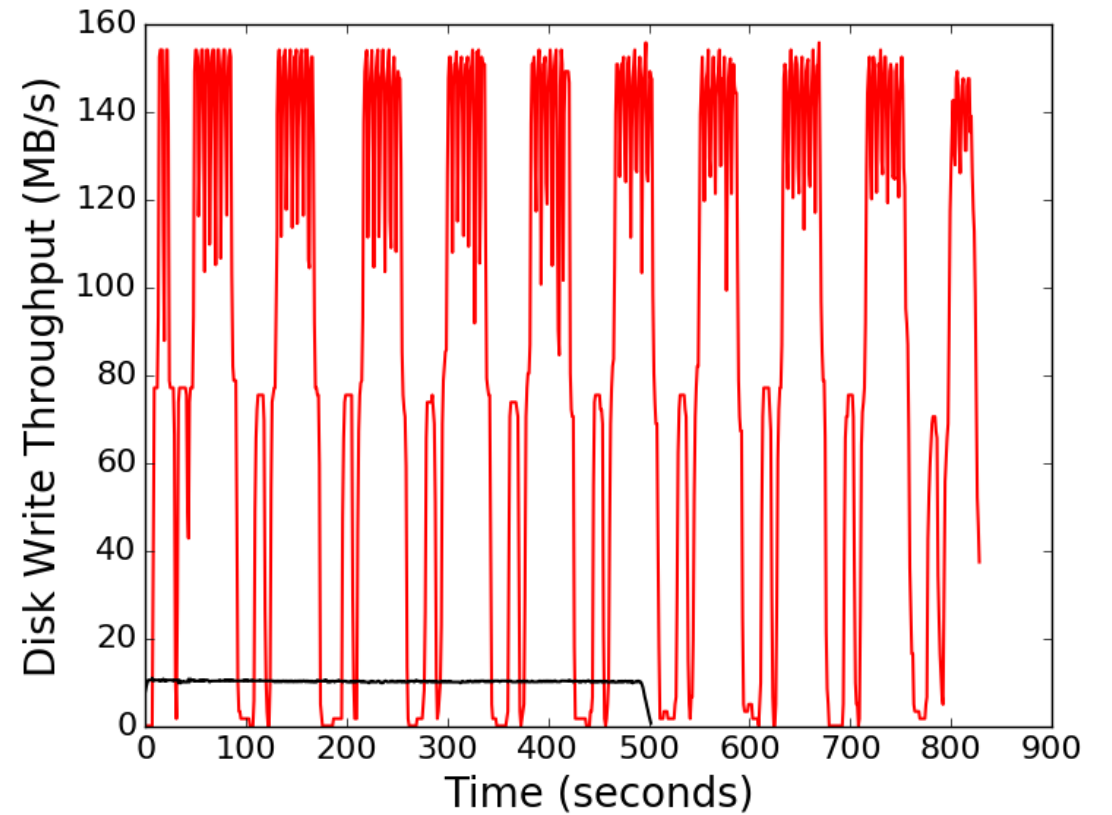
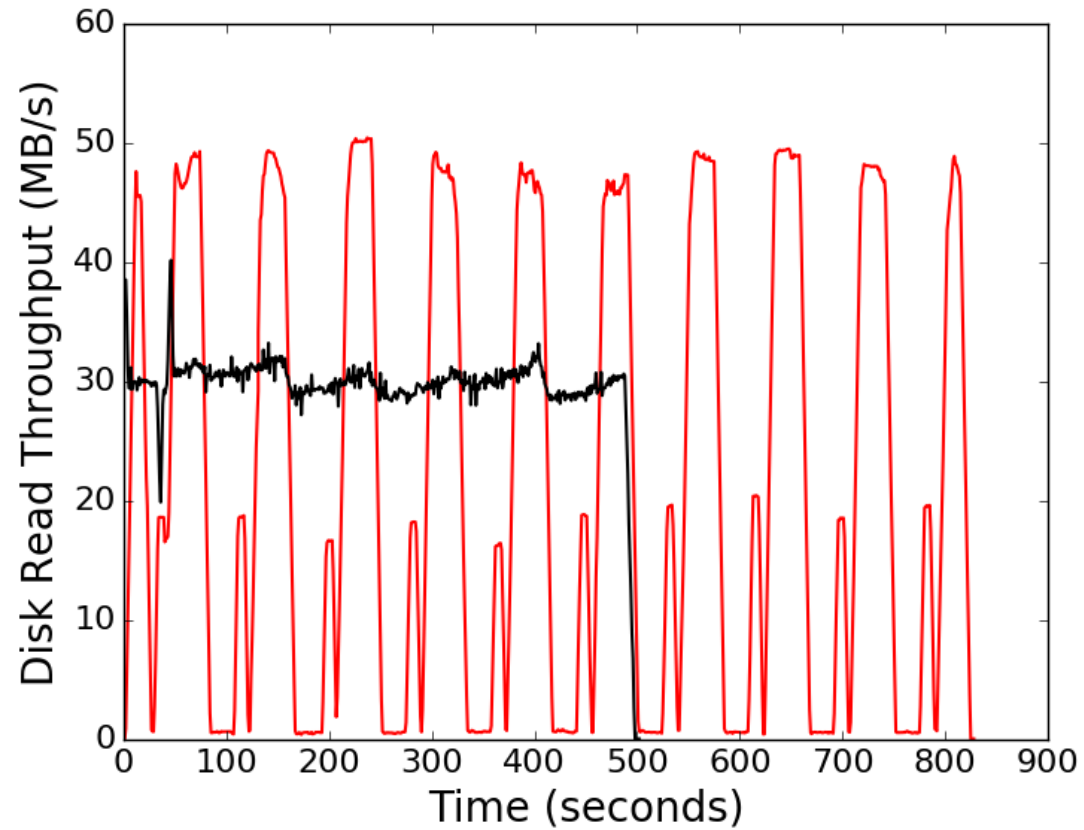
Performance – Sort and Dup. Mark

- Sort
 - By metadata or aligned location
 - 1.54x speedup over samtools
 - 5.15x speedup over Picard
- Dataset stats 
 - 2x speedup
- Duplicate marking
 - Same algorithm as samblaster
 - 3.73x faster than samblaster
- Coverage (depth) 
 - 2x speedup

Profiling

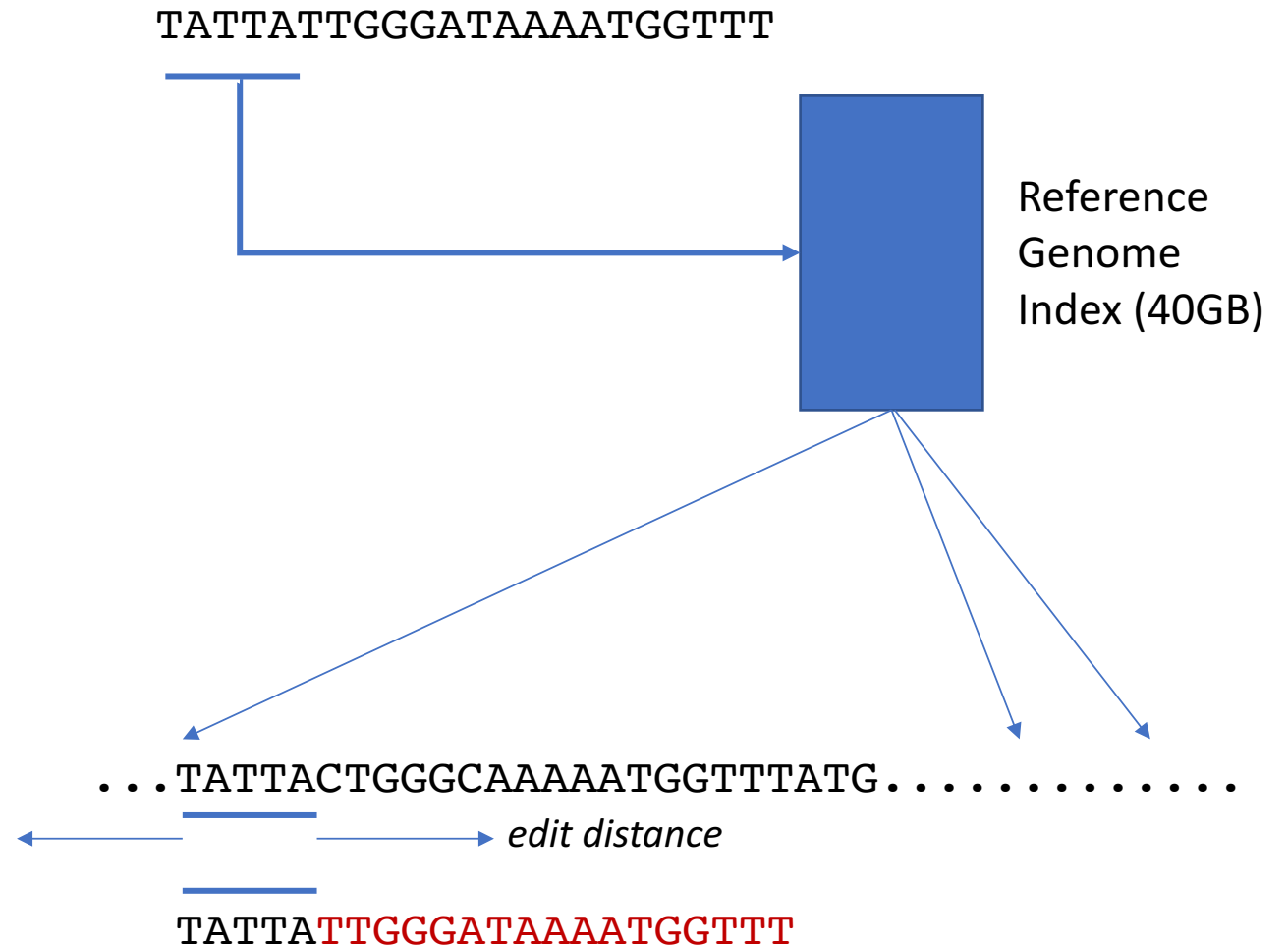


Read/Write Single Disk



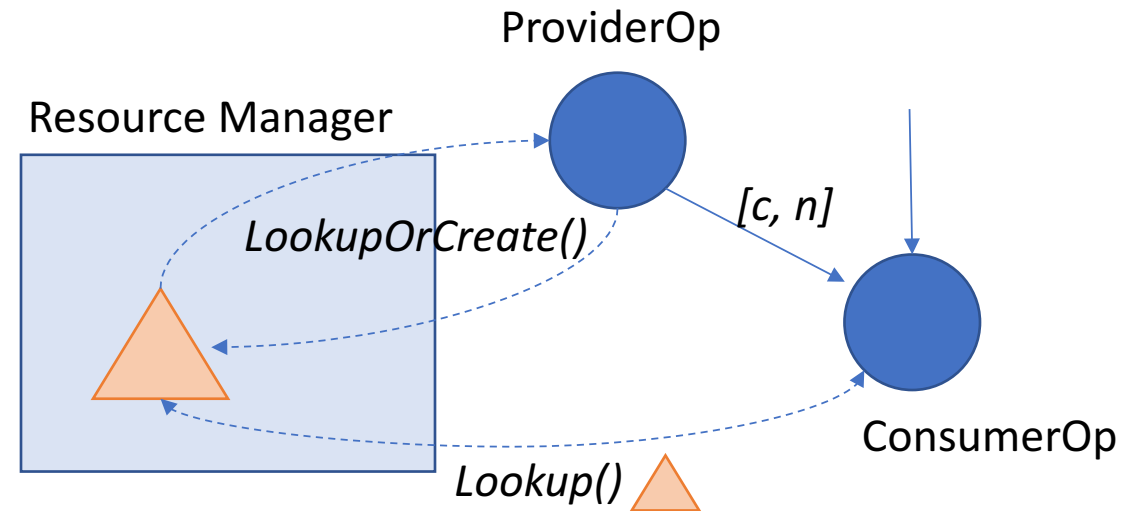
Alignment

- Example: SNAP
- Build hash index of reference
- To align a read:
 - Hash a portion (seed)
 - Lookup
 - Evaluate each hit
 - Edit distance computation
- Cores align reads in parallel



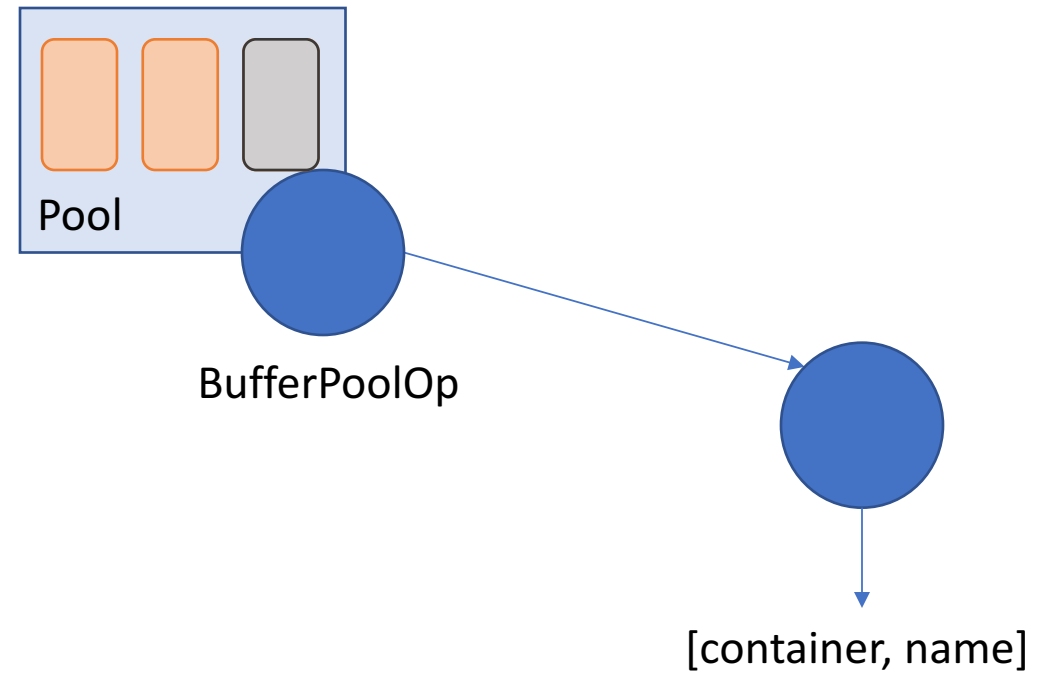
Shared Data

- Sometimes need to share data between ops
 - E.g. multi-GB index of reference genome
- Use TF session resource manager
 - [string, string] → refcount object
- Op can create objects, provide handle to other ops



Data Movement

- Tensors not amenable to bioinfo data
- Leverage TF shared resources
- Implement reusable buffers
 - Stable memory use
 - Avoid syscalls



Bioinformatics?

- Biology, computer science, math, statistics
- Started mid 90's with Human Genome Project
- Broad field
 - Genomics, proteomics, systems biology
- This talk: Whole Genome Sequence (WGS) analysis
 - Reading the letters of your DNA (ATCG ...)