

A Spark Framework for $< \$100$, < 1 Hour, Accurate Personalized DNA Analysis at Scale

Zaid Al-Ars

Delft University of Technology

The Netherlands



COST PER GENOME

2



National Institutes
of Health

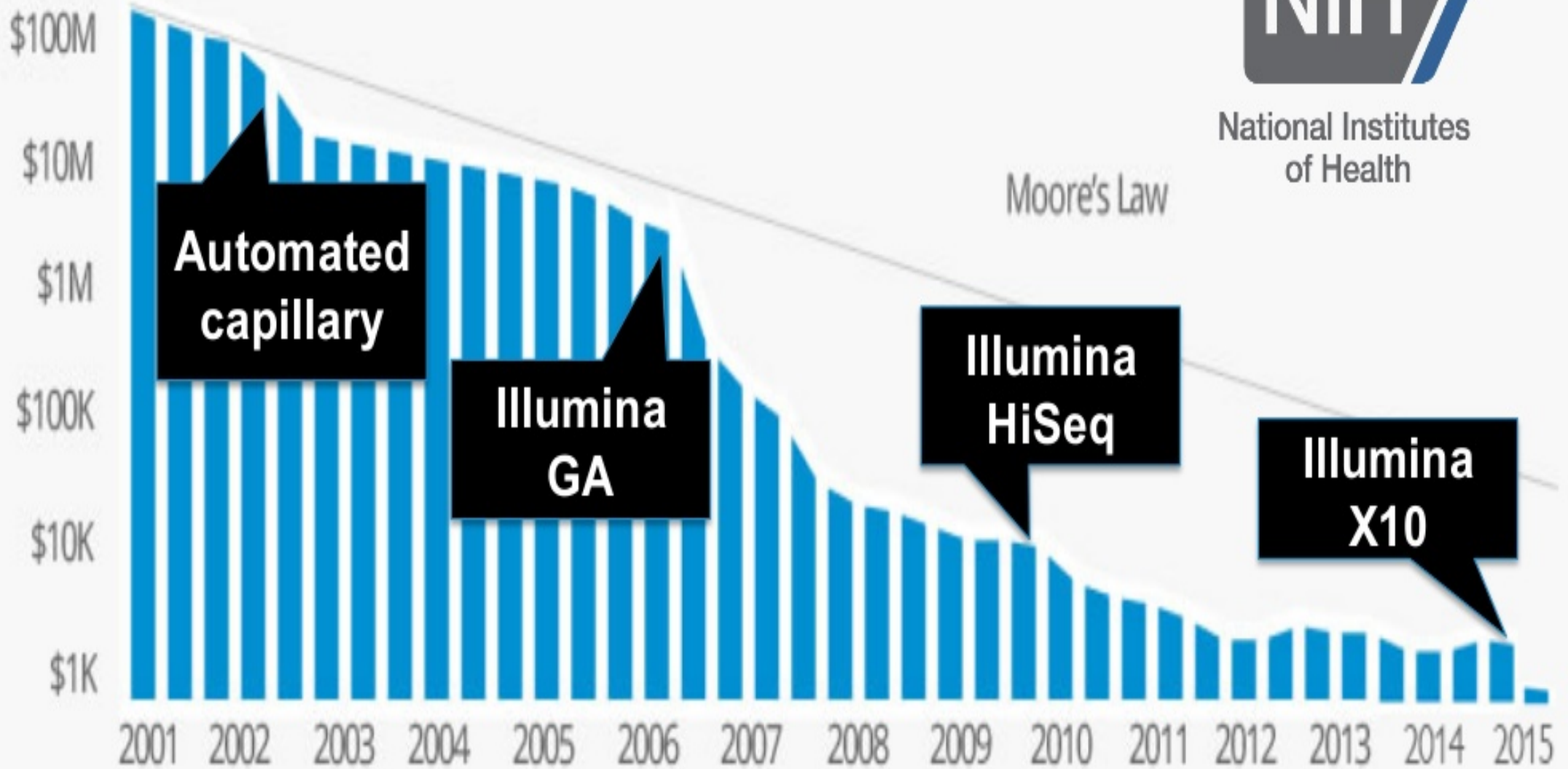
Moore's Law

**Automated
capillary**

**Illumina
GA**

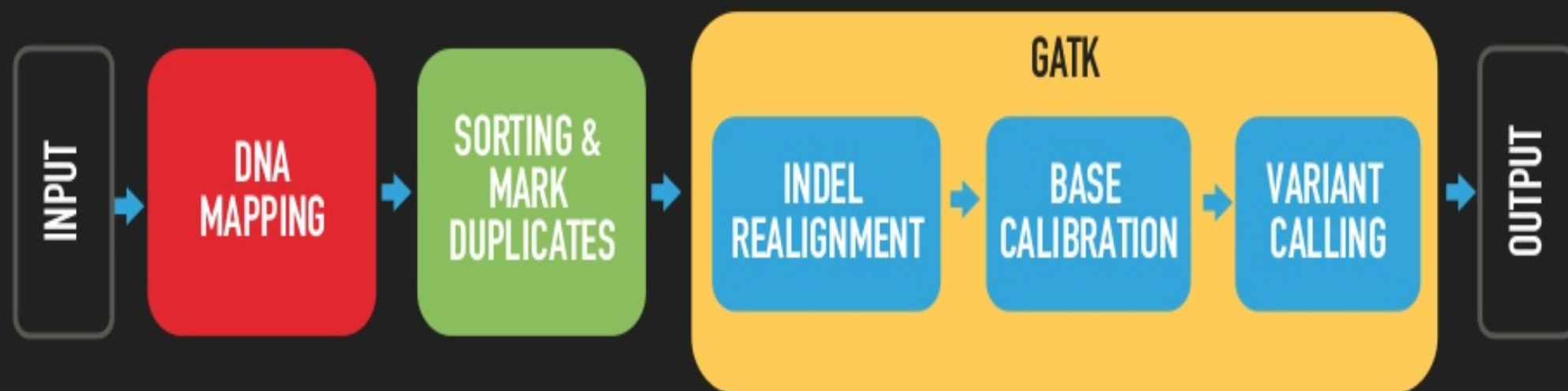
**Illumina
HiSeq**

**Illumina
X10**



DNA SEQUENCING COSTS 10^5 LESS THAN 15 YEARS AGO!

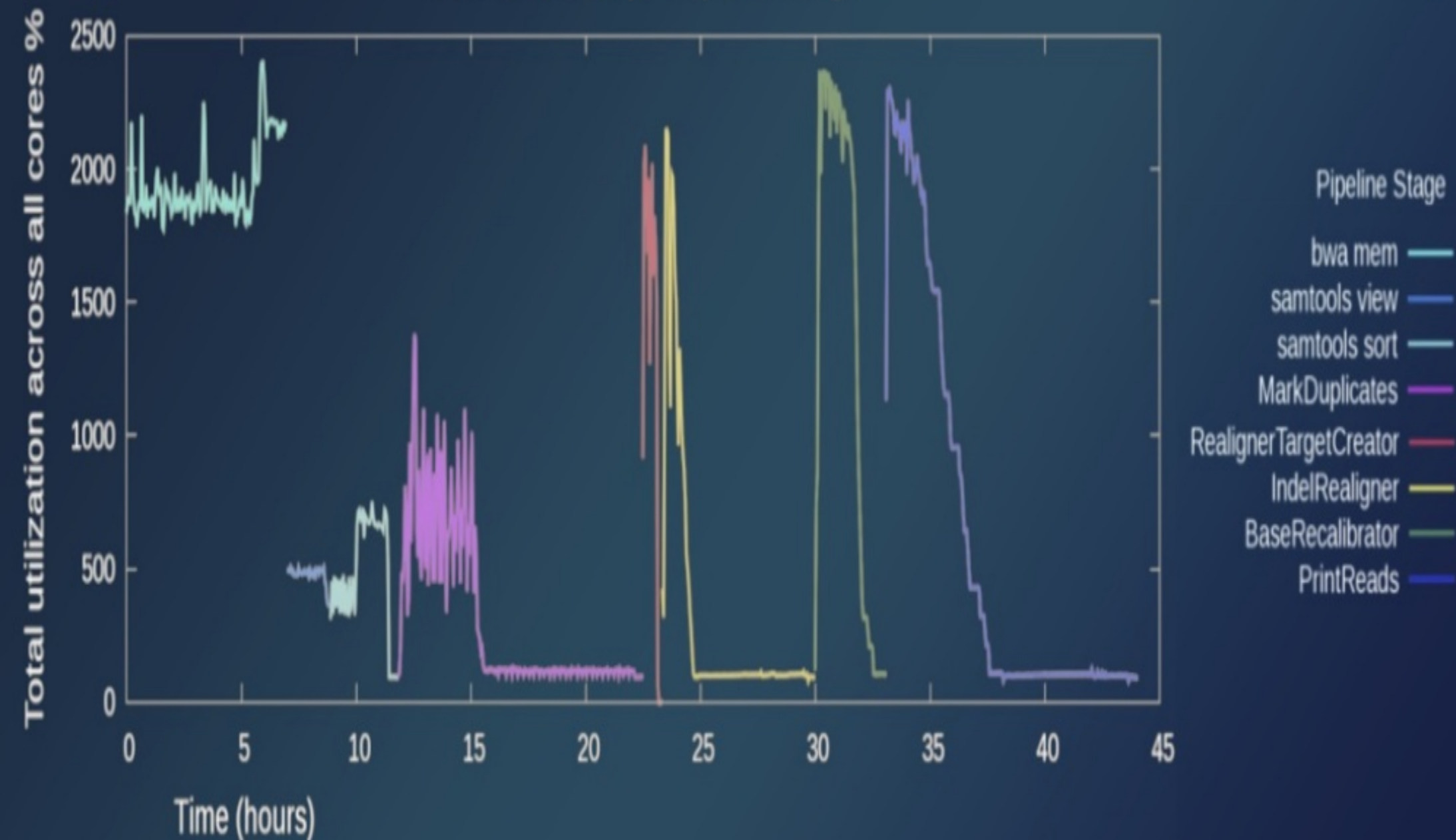




**STATE-OF-THE-ART DNA VARIANT DISCOVERY IS
GATK USES A PIPELINE OF DIFFERENT TOOLS**

Total Utilization per Pipeline Stage

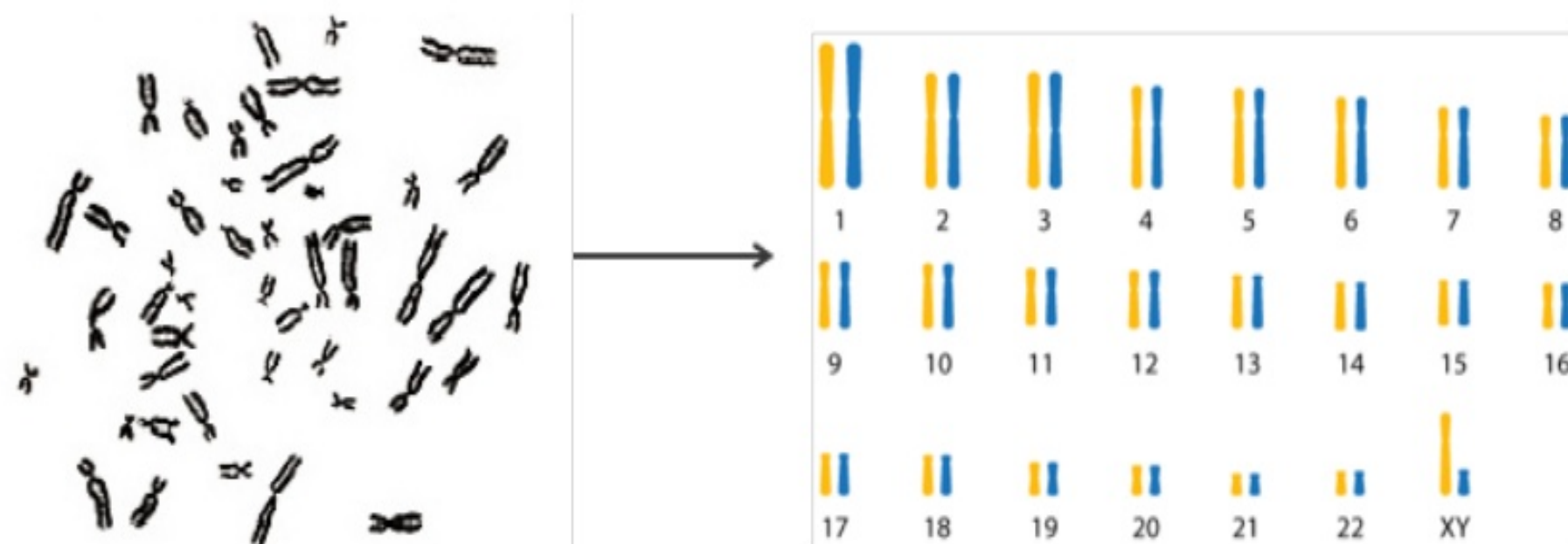
5



EXISTING PIPELINES DON'T EFFICIENTLY USE COMPUTERS

IDEA

- ▶ We use input data segmentation to group data of the same chromosome for parallelization.
- ▶ Data of the same chromosome can further be divided to create more segments.



COMPARISON OF BIG DATA TECHNIQUES

GATK Queue

- Slow scheduling
- Hard to setup
- Disk intensive
- Poorly documented



Halvade

- Disk intensive
- Limited memory use
- Static load balancing



Our Tool

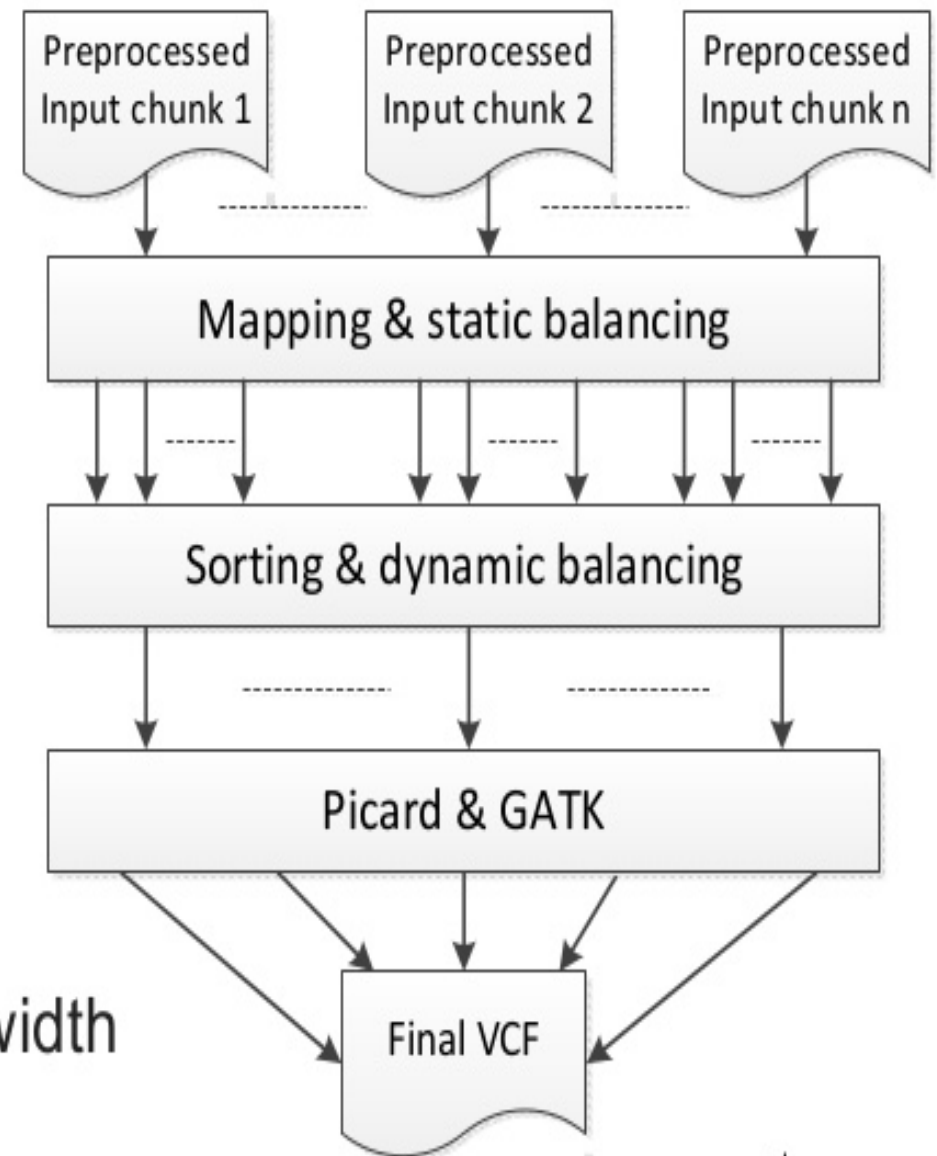
- ✓ In-memory compute
- ✓ Dynamic load balancing
- ✓ Effective use of disk
- ✓ Easy to setup



Lack of genomics verification of the outputs

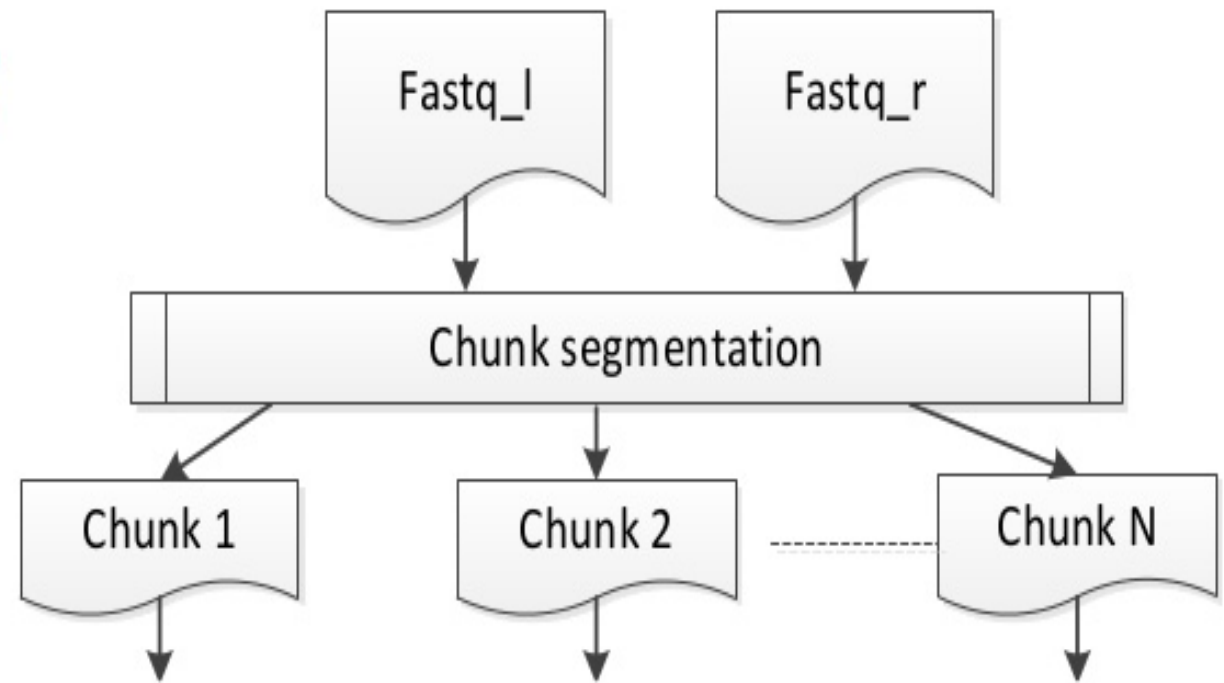
IMPLEMENTATION

- ▶ Input is preprocessed into chunks
- ▶ Spark framework divided into 3 stages
 1. Mapping & static load balancing
 2. Sorting & dynamic load balancing
 3. Picard & GATK
- ▶ Challenges in memory capacity, processor utilization and network bandwidth



PREPROCESSING INPUT

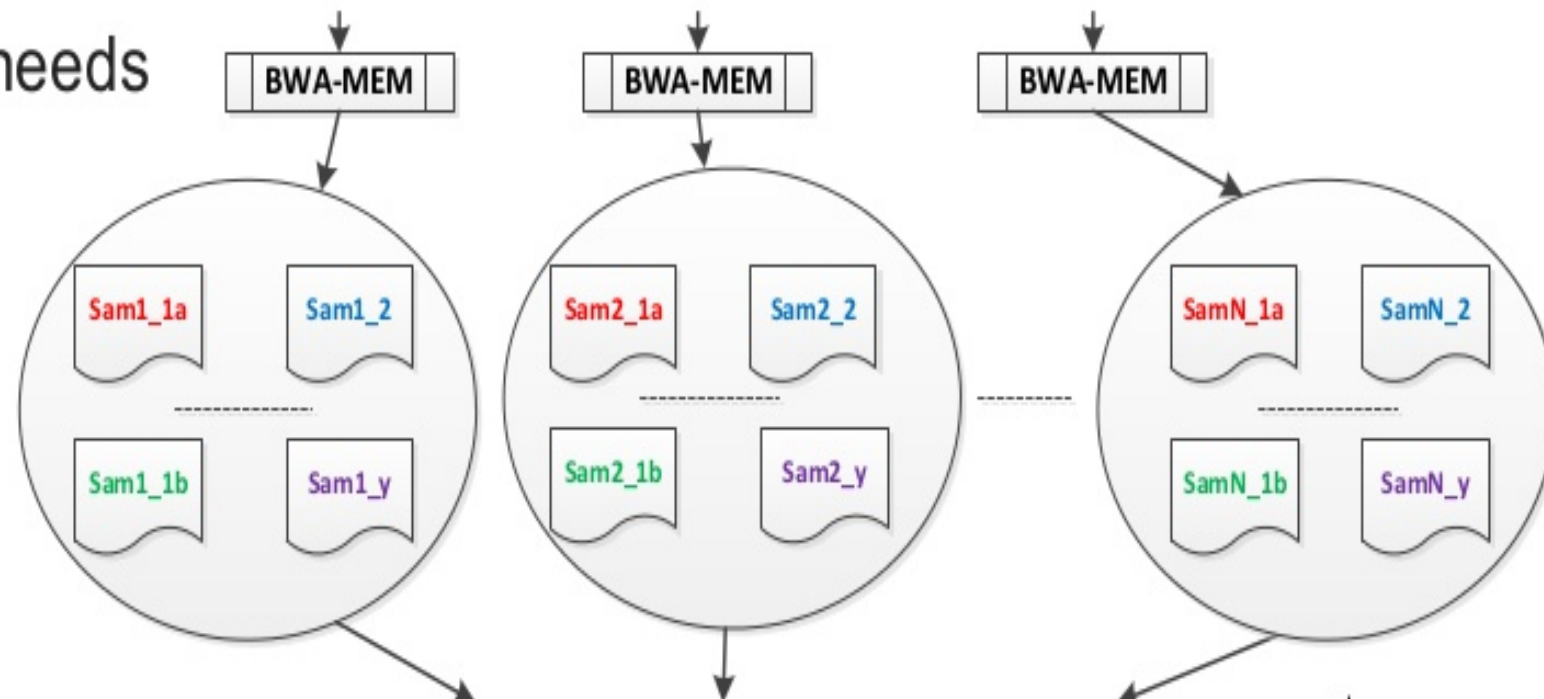
- ▶ Two input files are merged
- ▶ Divided into chunks
- ▶ No. of chunks depends on cluster size
- ▶ Performed once before data is processed
- ▶ Not included into pipeline timing calculation



1. MAPPING & STATIC LOAD BALANCING

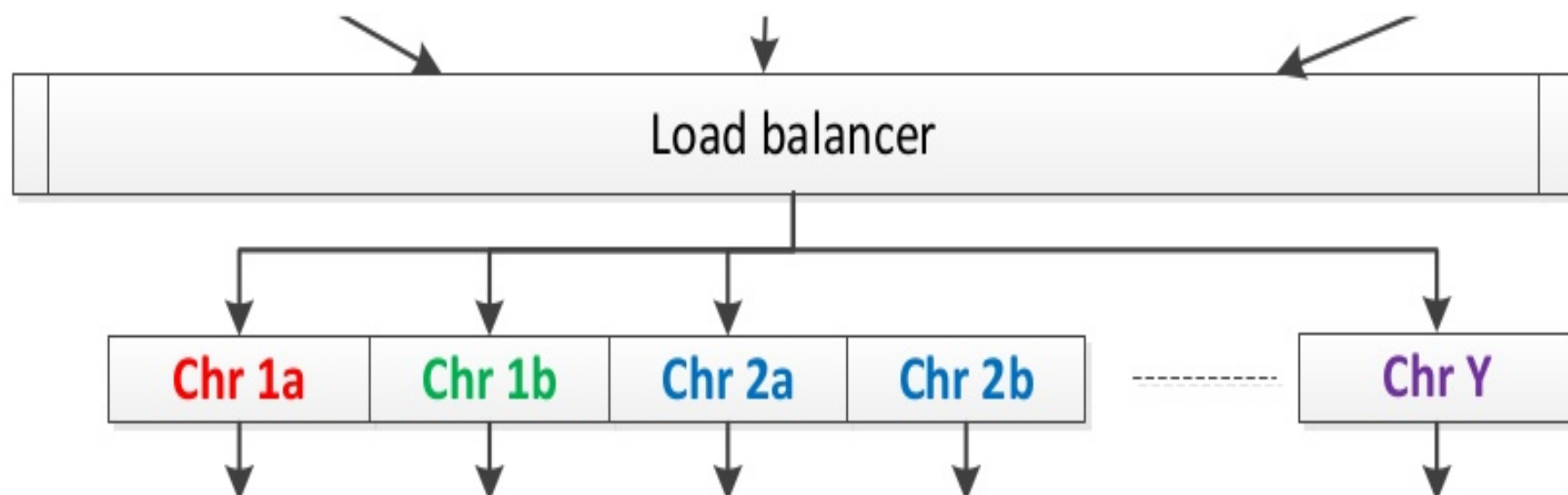
- ▶ Chunks mapped w/ BWA => easily scalable to all cores
- ▶ Output divided into chromosomal regions
- ▶ Reduces memory needs

for dynamic
load balancing



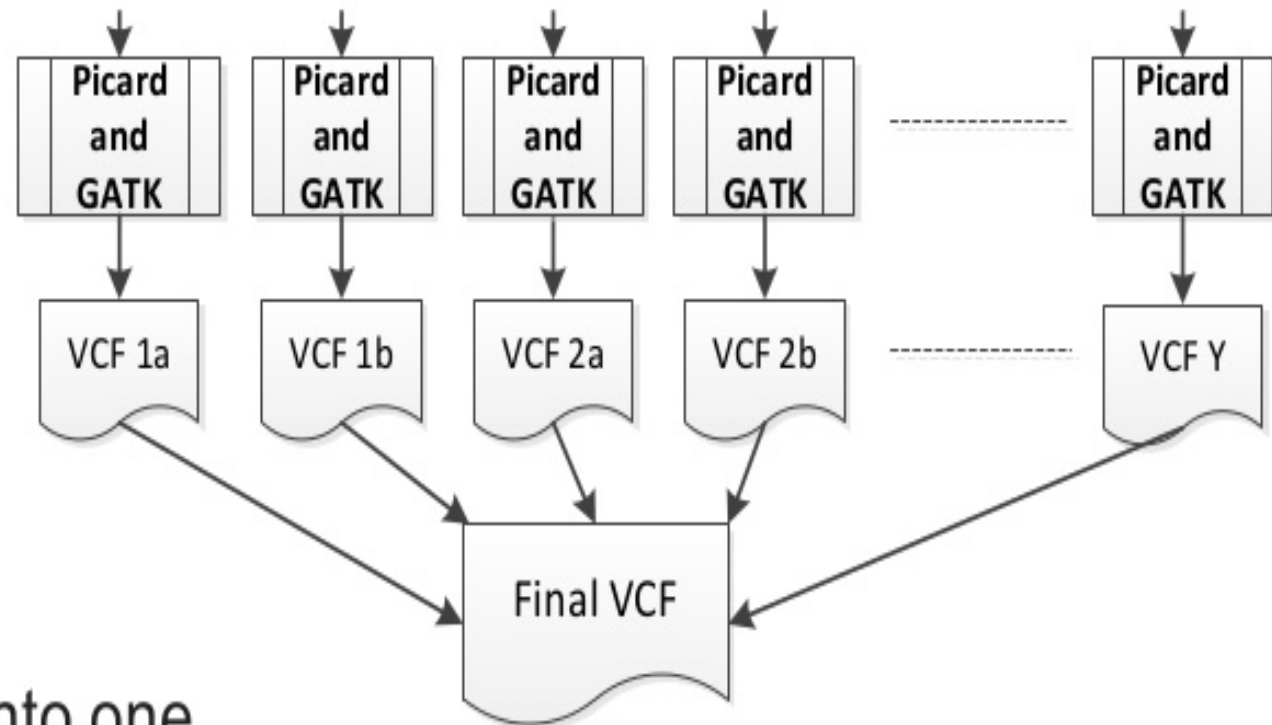
2. SORTING & DYNAMIC LOAD BALANCING

- ▶ Files of chromosomal regions are combined
- ▶ Files subdivided depending on # reads to ensure optimal load balancing



3. PICARD DEDUPLICATION & GATK VARIANT CALLING

- ▶ Many mini GATK pipelines run on region files
- ▶ # Spark containers depends on efficient core utilization
- ▶ Output VCF files combined into one output VCF file



EXPERIMENT SETUP

POWER8 Cluster

- 20 POWER8 S822LC nodes, bare metal
- 2x SCM 10-core, SMT8; 160 HW threads per node
- 512 GB of RAM per node
- Mellanox Infiniband EDR ConnectX-4 adapters (100Gb/s)
- IBM General Parallel File System (GPFS)

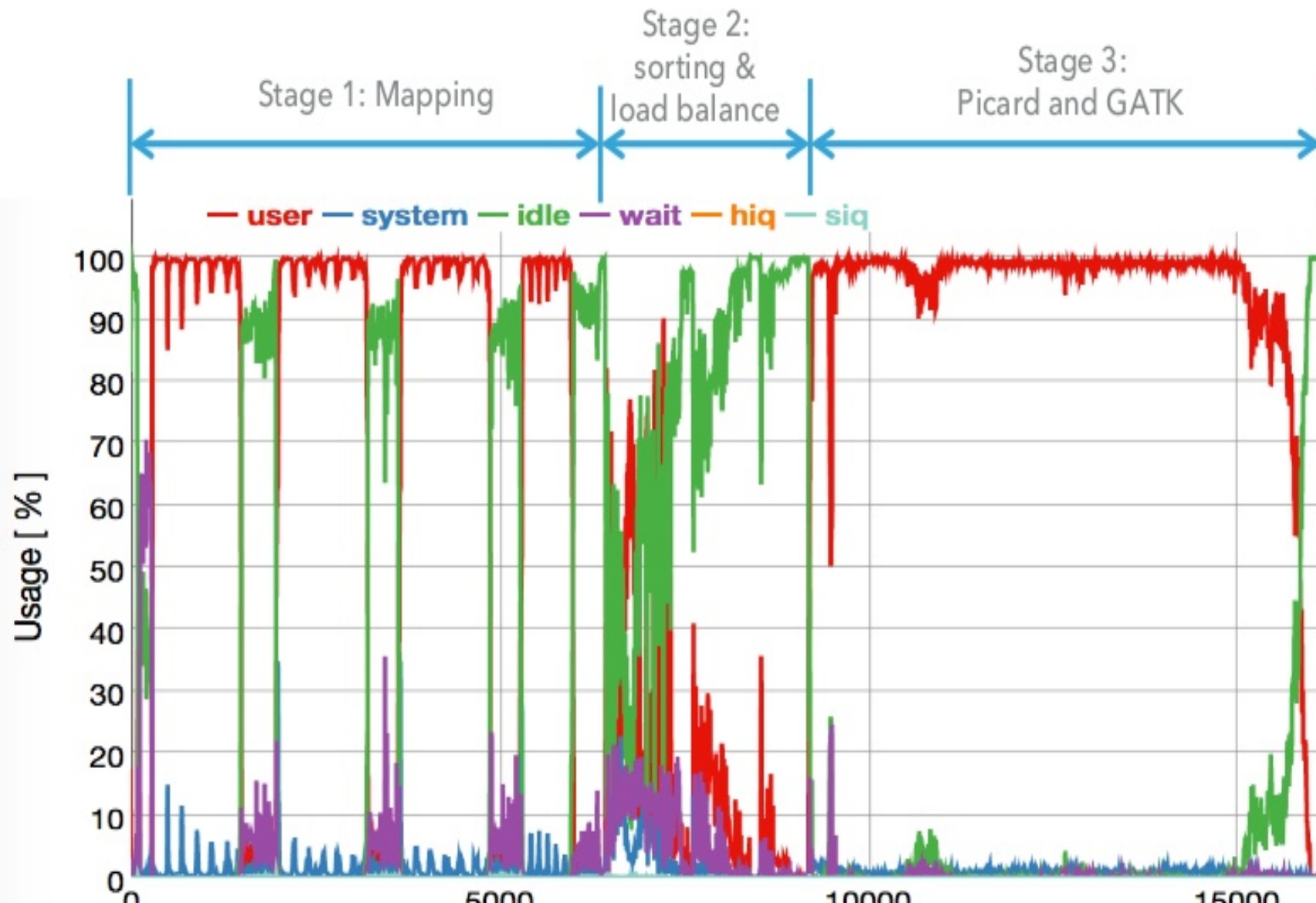
- Red Hat EL 7.2
- Spark 1.5.1, openJDK 1.8
- IBM LSF for resource management
- Configured as 1 master + 19 slaves

Data Used:

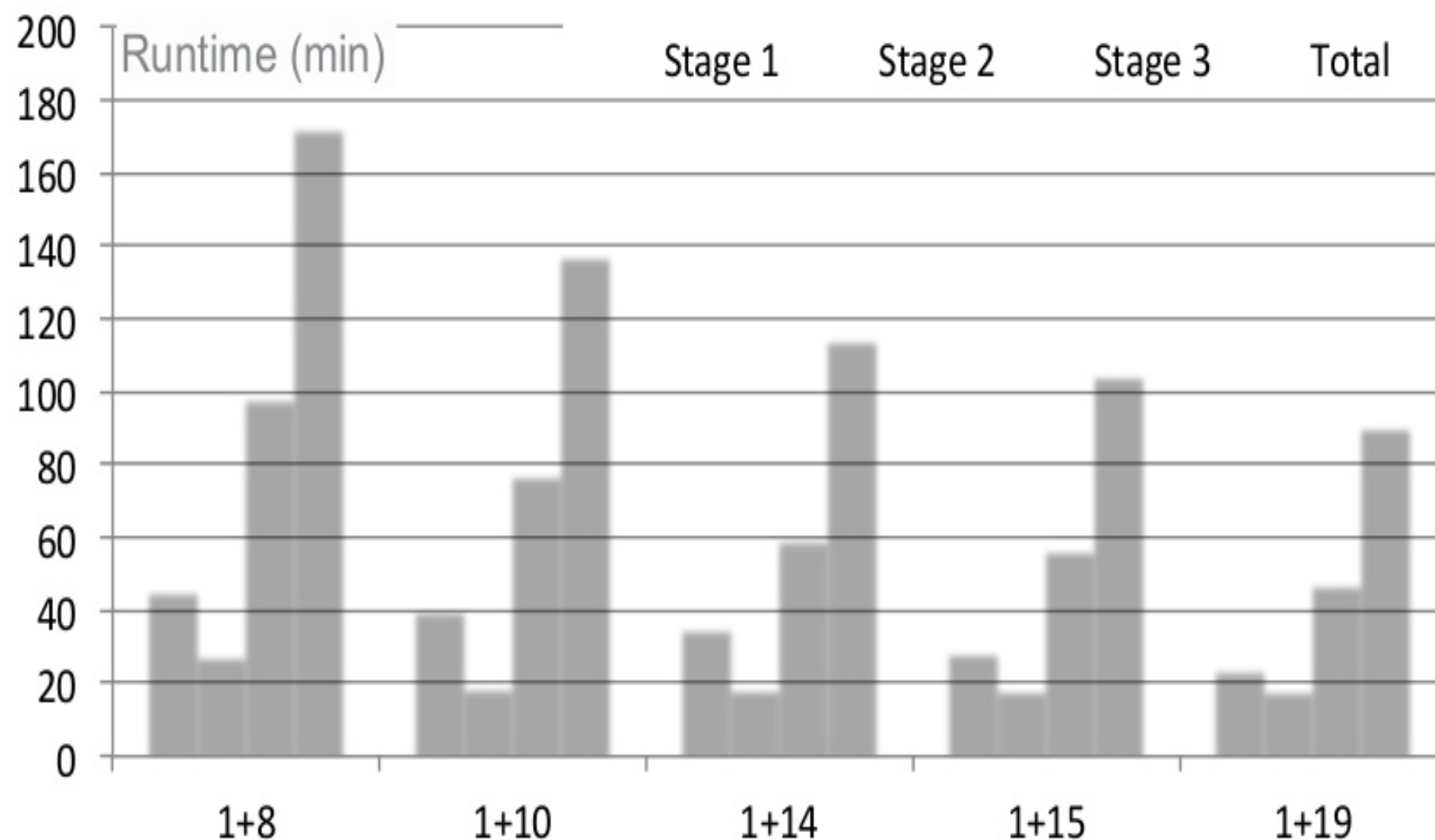
- Raw reads: G15512.HCC1954 (whole human genome, 400GB uncompressed)
- Reference: human g1k v37



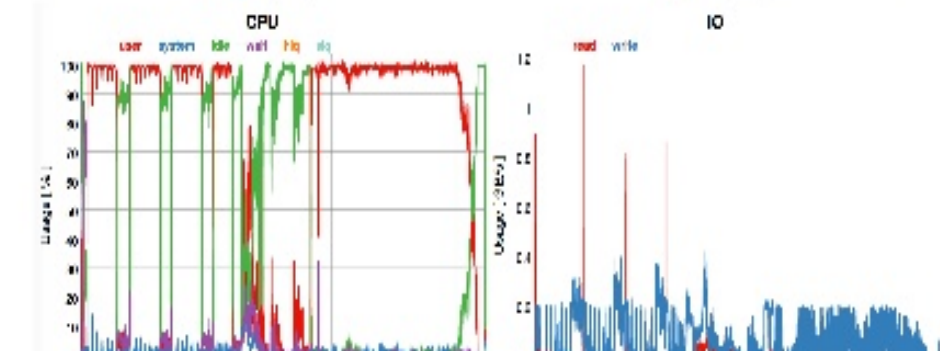
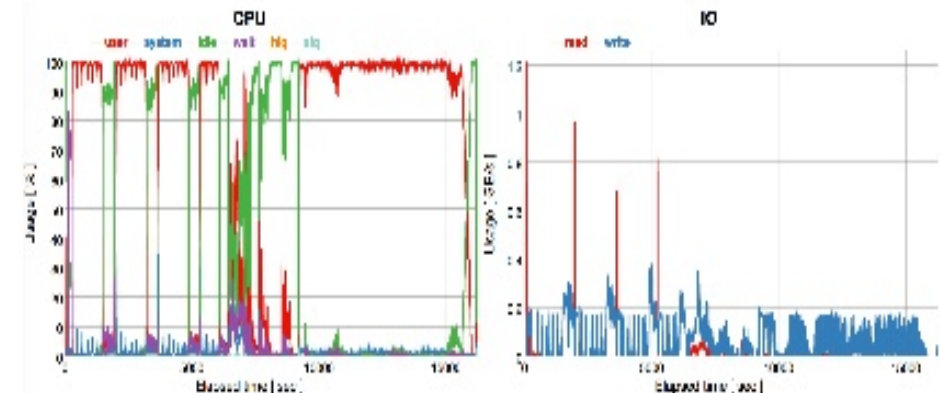
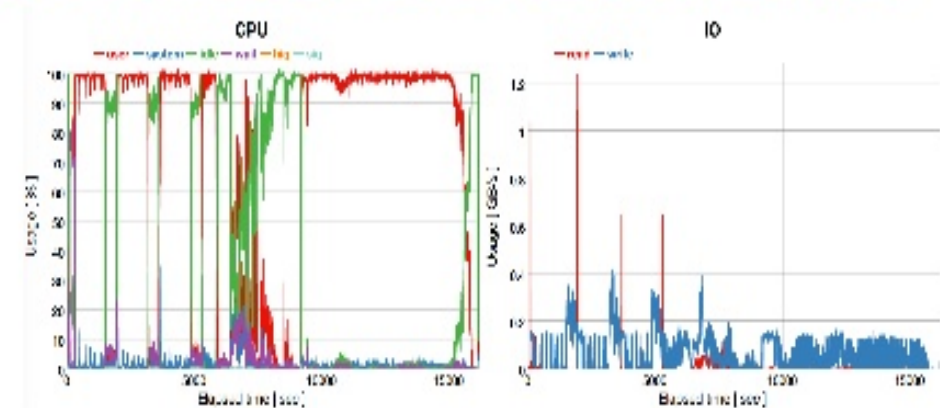
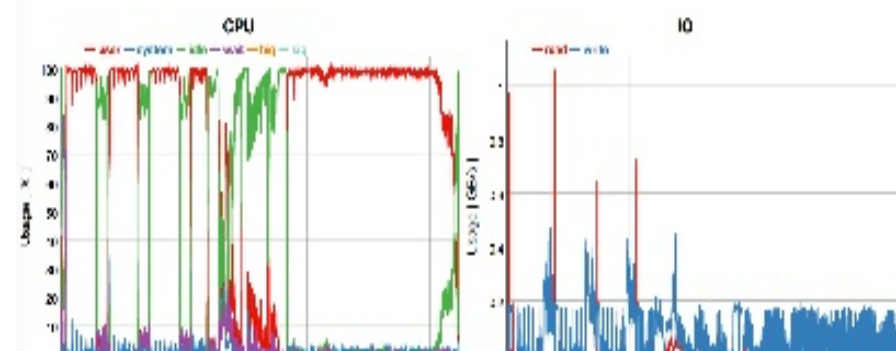
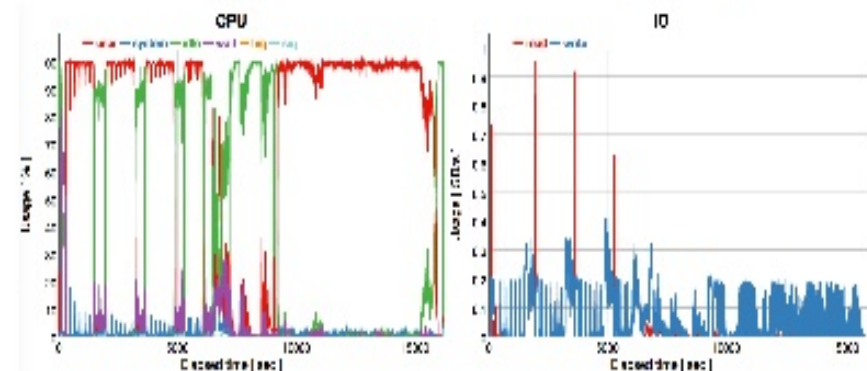
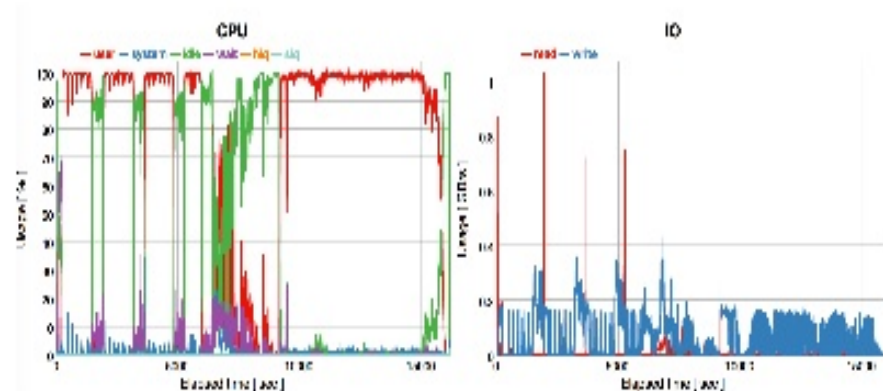
CPU UTILIZATION ON SMALLER CLUSTER



RUNTIME SCALABILITY ON 20-NODE POWER8 CLUSTER

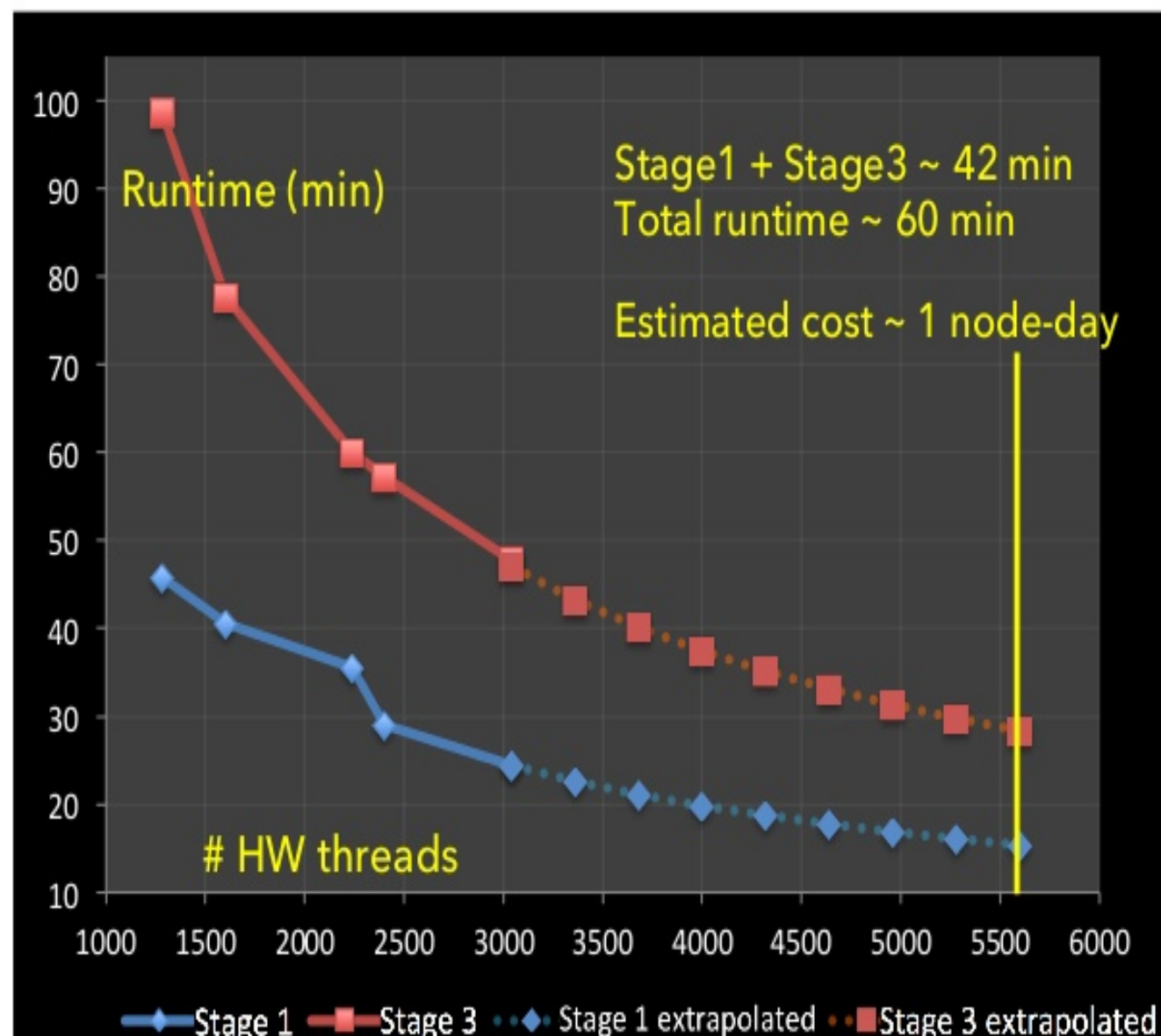


UNIFORM CPU UTILIZATION AND IO AMONG SIX NODES



SCALABILITY ANALYSIS FOR LARGER CLUSTERS

- ▶ 90 min runtime on 20-node cluster
- ▶ Stage 1 & 3 are scalable
- ▶ Runtime scales down to 1 hour for 35 node cluster



ACCURACY

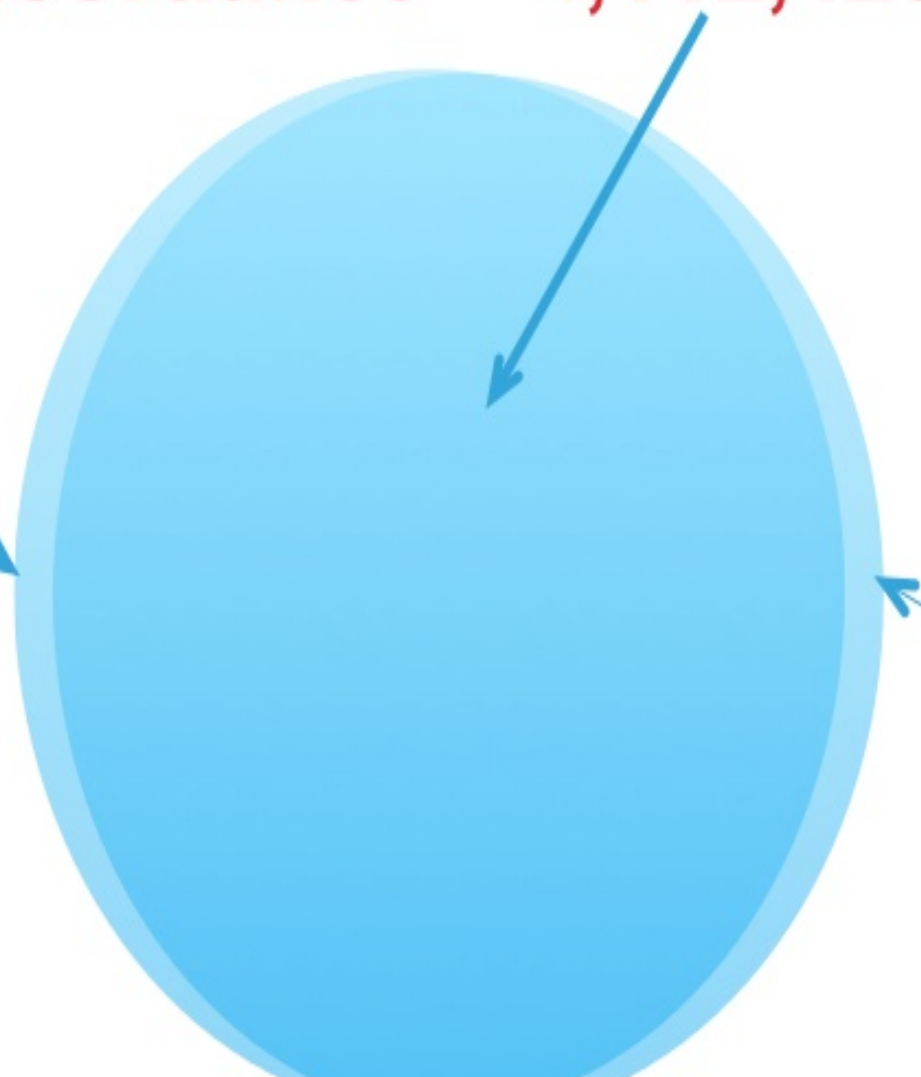
Concordance = 4,112,429 (99.1%)

Spark GATK:

38,052 (0.93%)

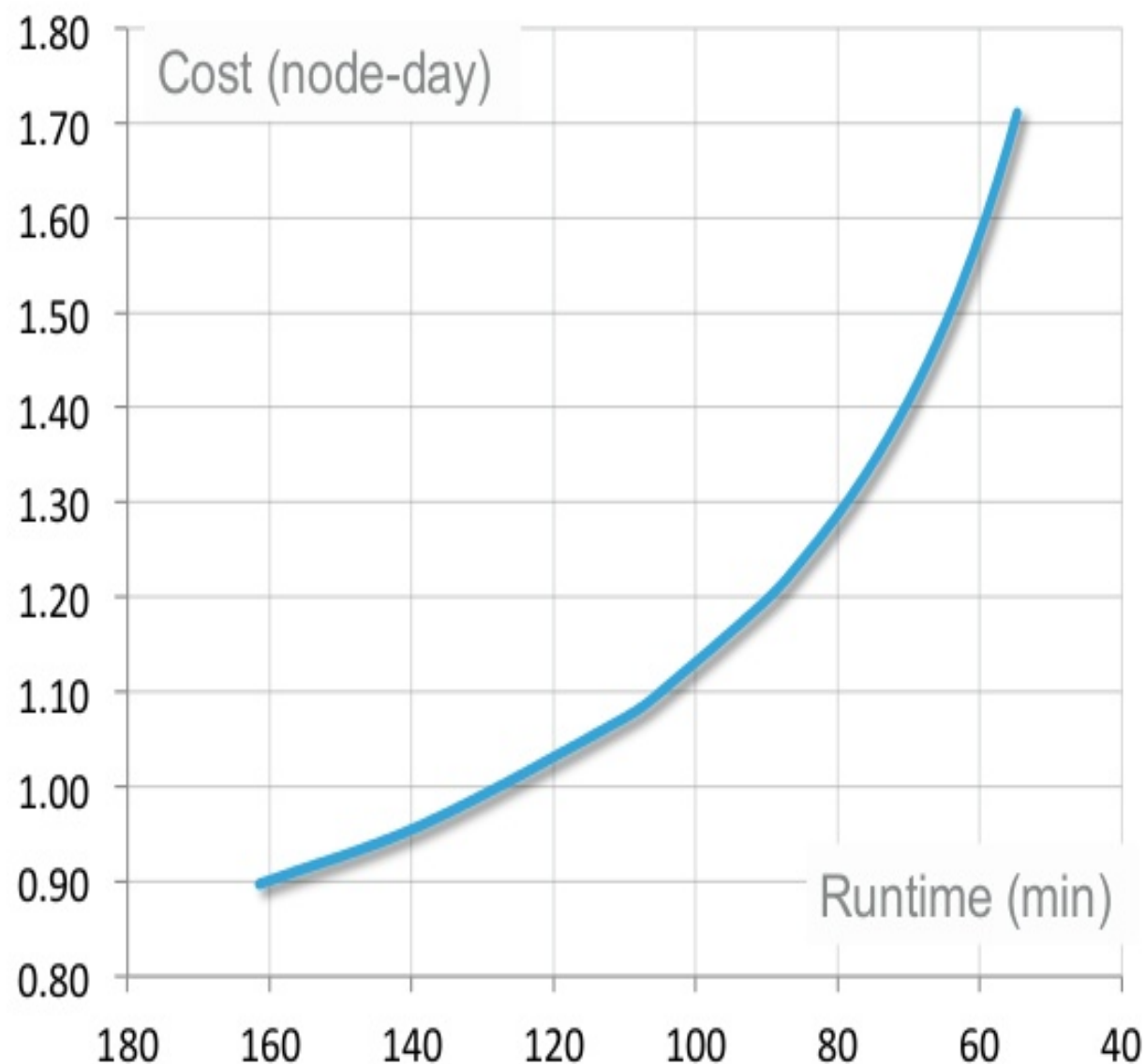
Serial GATK:

11,040 (0.27%)



COST ESTIMATE

- ▶ GATK Spark allows efficient (i.e., cheap) CPU utilization
- ▶ 60 minute runs on 35-nodes cluster
- ▶ POWER nodes cost <\$100 a day (\$30-\$70 in SoftLayer)



CONCLUSIONS

- ▶ GATK pipeline runs on Apache Spark framework
- ▶ Solution shows good scalability without sacrificing accuracy
- ▶ 400GB WGS data scales to under 1 hour for 35 nodes
- ▶ Price point is below \$100 for POWER8 cluster
- ▶ Future work
 - ▶ We are accelerating our solution further using FPGAs through the IBM POWER8's CAPI interface
 - ▶ Examples: compression & math-heavy kernels like pairHMM

ACKNOWLEDGEMENTS

- ▶ Hamid Mushtaq (TUDelft)
- ▶ Carlos Costa (IBM)
- ▶ Neil Graham (IBM)
- ▶ Peter Hofstee (IBM)
- ▶ Raj Krishnamurthy (IBM)
- ▶ Frank Liu (IBM)
- ▶ Gang Liu (IBM)
- ▶ Rei Odaira (IBM)
- ▶ Indrajit Poddar (IBM)

THANK YOU.

Zaid Al-Ars

z.al-ars@tudelft.nl

Delft University of Technology

The Netherlands

<http://ce.ewi.tudelft.nl/zaid>



SPARK SUMMIT 2016
DATA SCIENCE AND ENGINEERING AT SCALE