# Native GATK: why you should care about performance

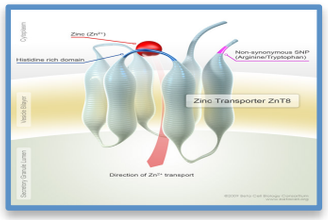
**Mauricio Carneiro, PhD**
Group Lead, Computational Technology Development
Broad Institute

Software is becoming slower more rapidly than hardware is becoming faster

–Niklaus Wirth, a plea for lean software

# The Importance of Scale…Early Success Stories (at 1,000s of exomes)

## Type 2 Diabetes

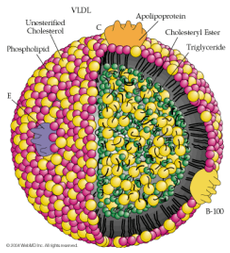



- 13,000 exomes
- SLC30A8
  (Beta-cell-specific $Zn^{++}$ transporter)
- 3-fold protection against T2D!
- ***1 LoF per 1500 people***

## Schizophrenia

- 5,000 exomes
- Pathways
  - Activity-regulated cytoskeletal (ARC) of post-synaptic density complex (PSD)
  - Voltage-gated $Ca^{++}$ Channel
- 13-21% risk in carriers
- ***Collection of rare disruptive mutations (~1/10,000 carrier frequency)***

## Coronary Heart Disease




- 3,700 exomes
- APOC3
- 2.5-fold protection from CHD
- ***4 rare disruptive mutations (~1 in 200 carrier frequency)***

## Early Heart Attack

- 5,000 exomes
- APOA5
- 22% risk in carriers
- ***0.5% Rare disruptive / deleterious alleles***

# Broad Institute in 2013

**50**
HiSeqs

**10**
MiSeqs

**2**
NextSeqs

**14**
HiSeq X

**6.5**
Pb of data

**427**
projects

**180**
people

**2.1**
Tb/day

* we also own 1 *Pacbio RS* and 4 *Ion Torrent* for experimental use

# Broad Institute in 2013

**44,130**
exomes

**2,484**
exome express

**2,247**
genomes

**2,247**
assemblies

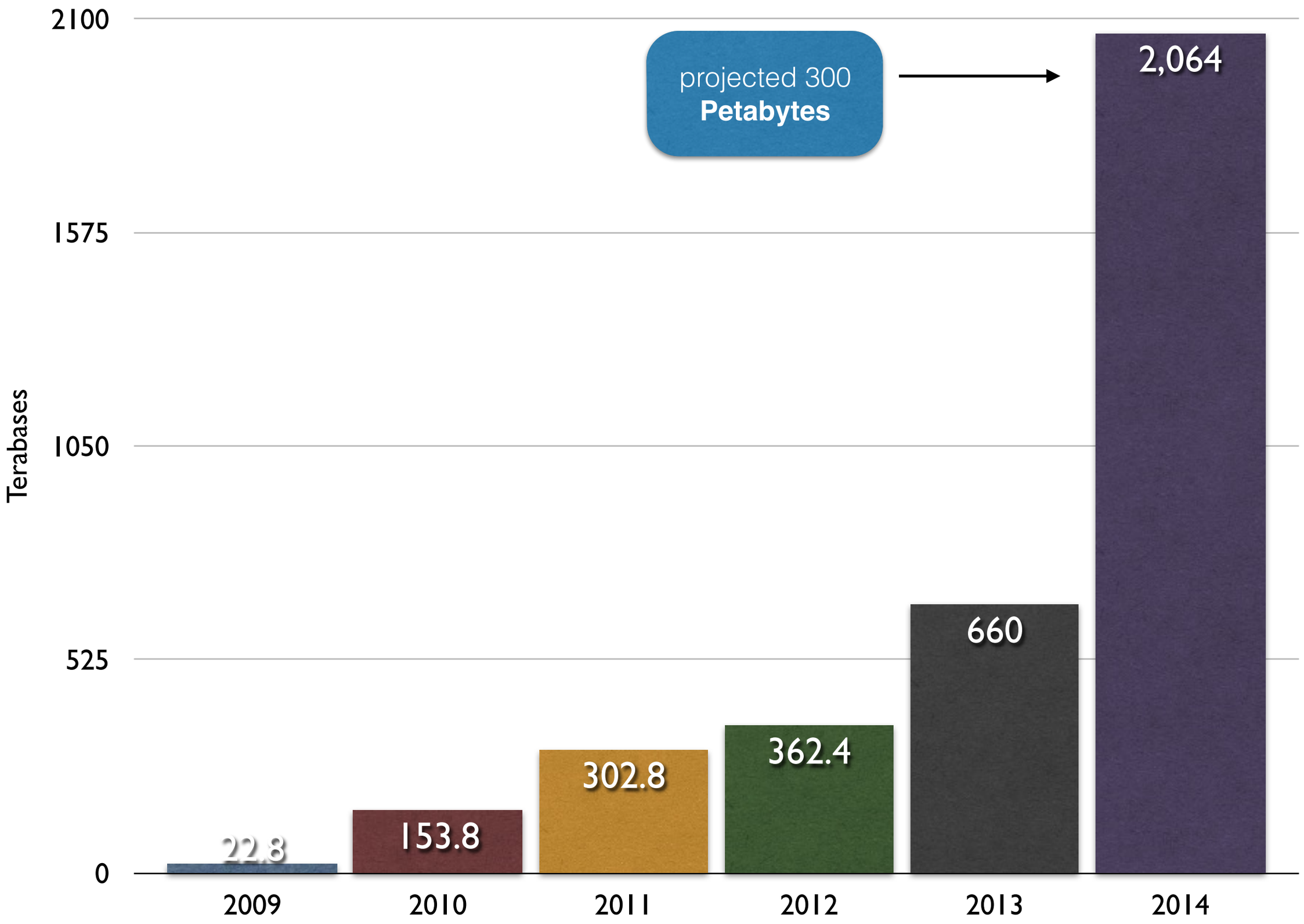
**8,189**
RNA

**9,788**
16S

**47,764**
arrays

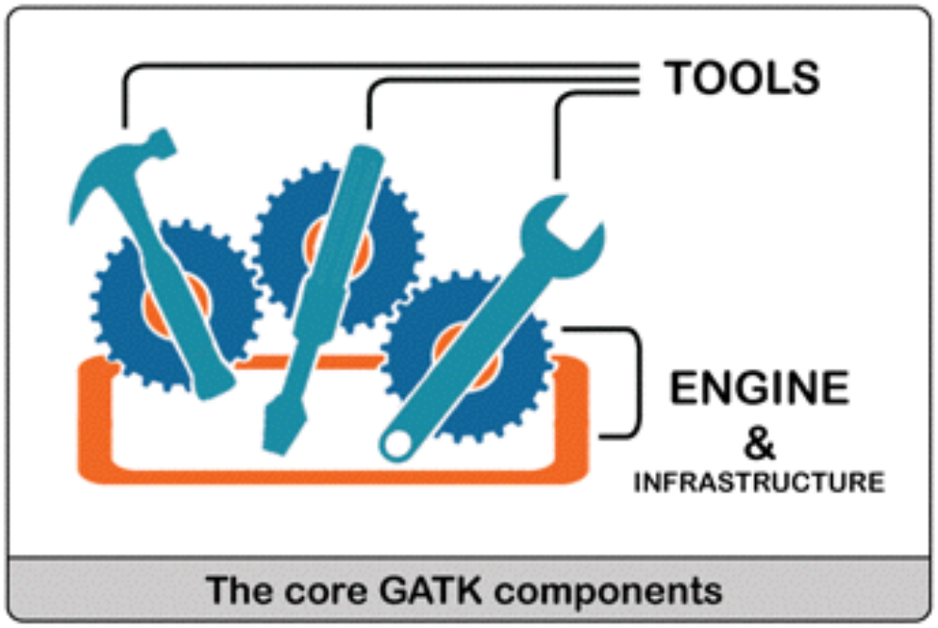
**228**
cell lines

# Terabases of Data Produced by Year



# GATK is both a toolkit and a programming framework, enabling NGS analysis by scientists worldwide

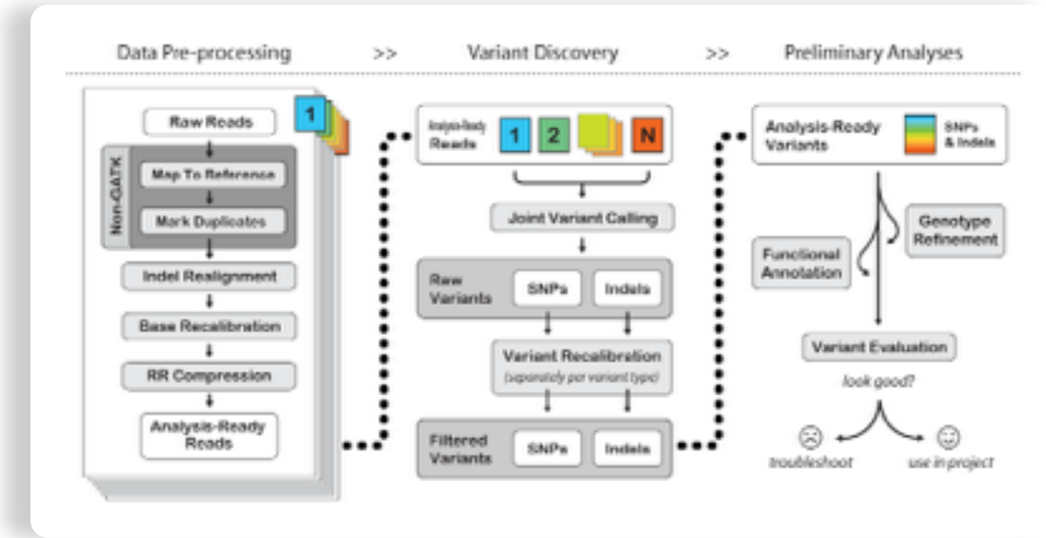
## Toolkit & framework packages



The core GATK components

Toolkit

*Best practices for variant discovery*




Framework

MuTect, XHMM, GenomeSTRiP, ...

*Tools developed on top of the GATK framework by other groups*

## Extensive online documentation & user support forum serving >10K users worldwide


http://www.broadinstitute.org/gatk

**About**
Overview of the GATK and the people behind it

**Guide**
Detailed documentation, guidelines and tutorials

**Community**
Forum for questions and announcements

**Events**
Materials from live and online events

iSeqTools

# Workshop series educates local and worldwide audiences

**Past:**

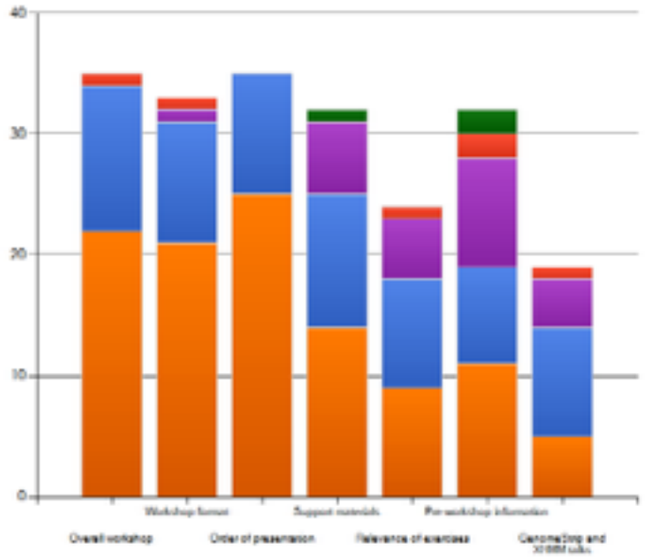
- Dec 4-5 2012, Boston
- July 9-10 2013, Boston
- July 22-23 2013, Israel
- Oct 21-22 2013, Boston
- March 3-5 2014, Thailand
- June 6-9 2014, Belgium
- Sep 17-18 2014, Philadelphia
- Oct 18-29 2014, San Diego

**Format**

- Lecture series (general audience)
- Hands-on sessions (for beginners)

**Portfolio of workshop modules**

- GATK Best Practices for Variant Calling
- Building Analysis Pipelines with Queue
- Third-party Tools:
  - GenomeSTRiP
  - XHMM




- High levels of satisfaction reported by users in polls
- Detailed feedback helps improve further iterations


**Tutorial materials, slide decks and videos all available online through the GATK website, YouTube and iTunesU**



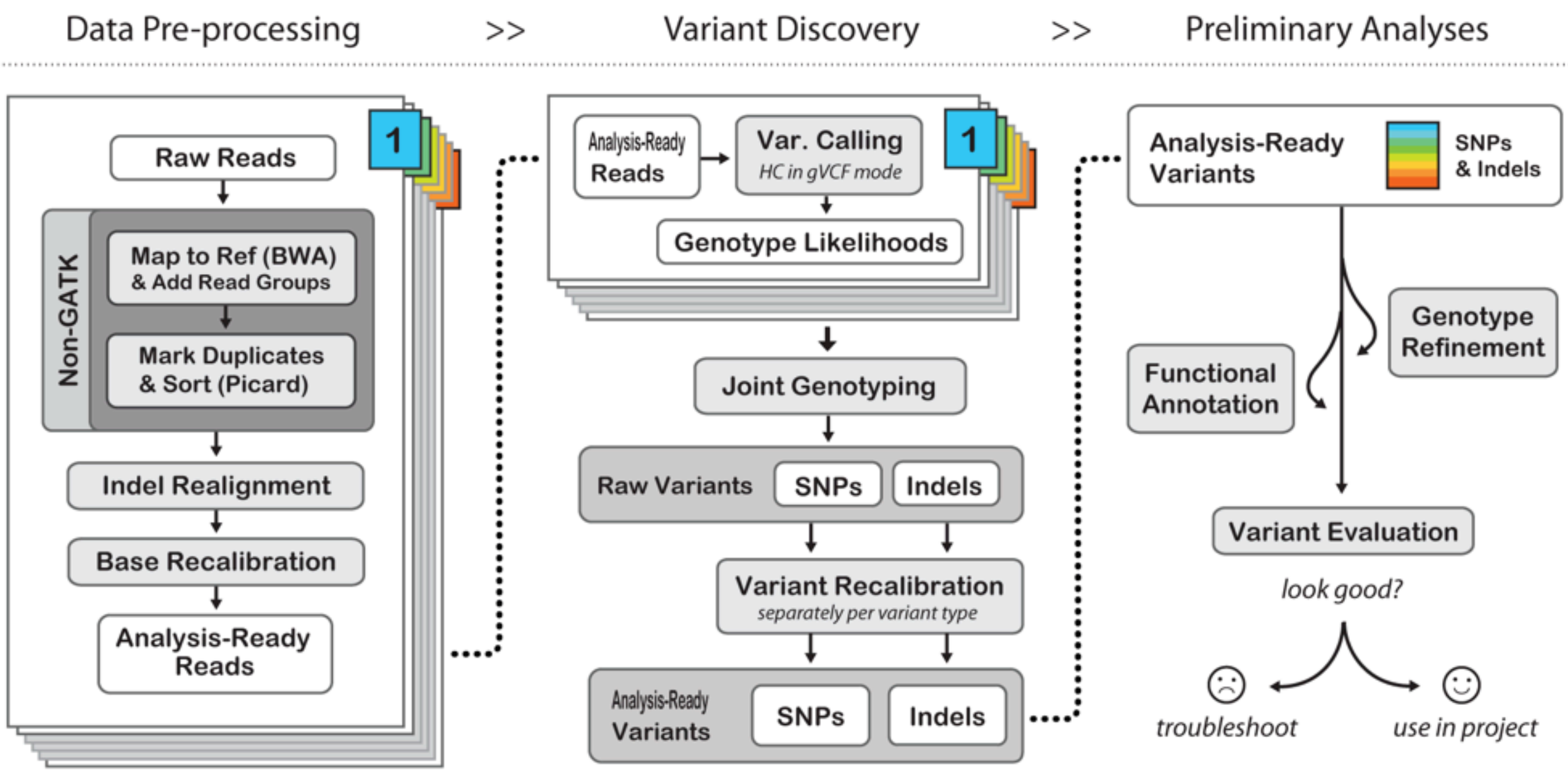
**BroadE: Overview of GATK & best practices**

by **broadinstitute** • 1 week ago • 1 view

Copyright Broad Institute, 2013. All rights reserved. The presentations below were filmed during the 2013 **GATK** Workshop, part of ...

NEW HD

# We have defined the best practices for sequencing data processing



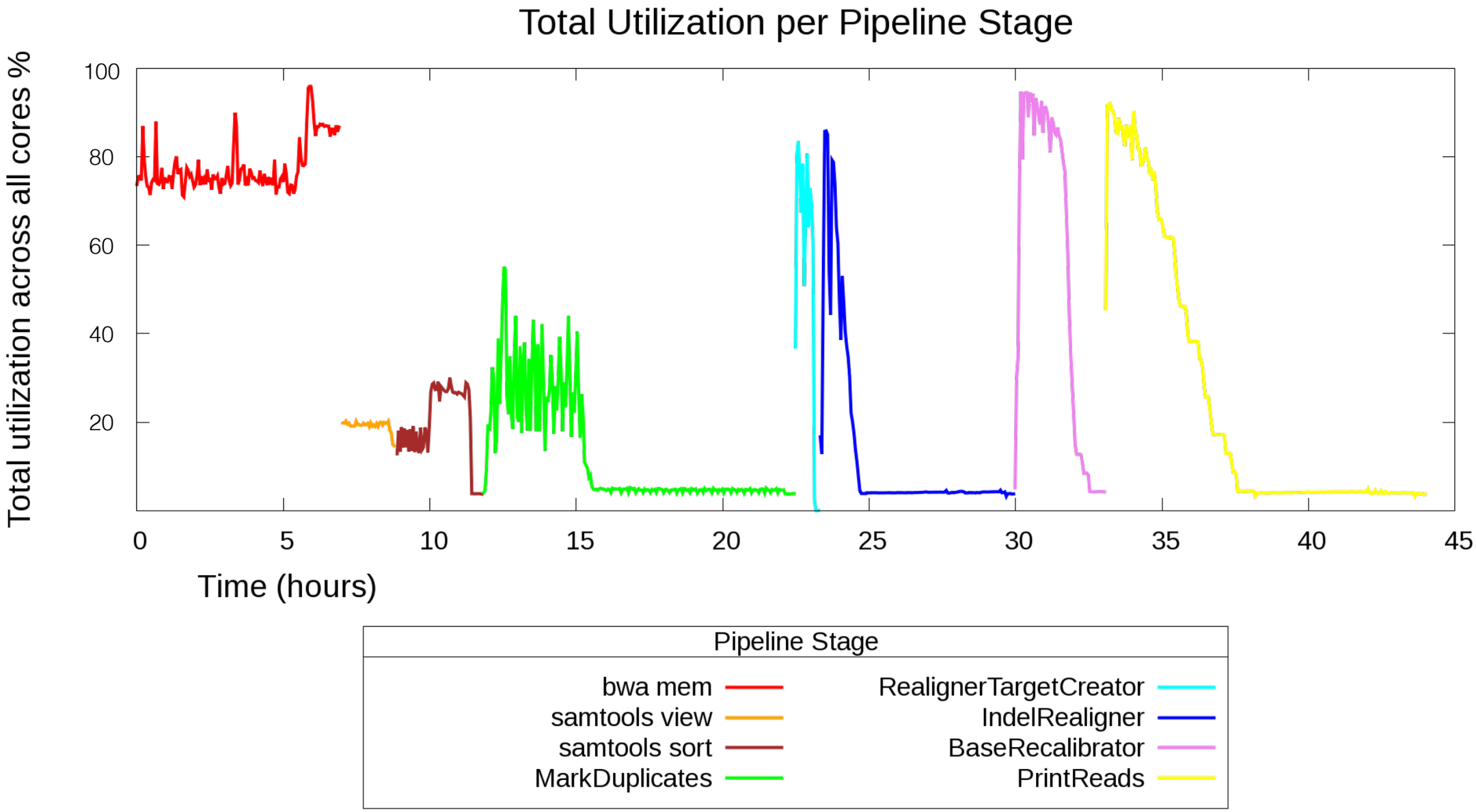

Auwera, GA et al. *Current Protocols in Bioinformatics* (2013)

# It takes 2 days to process a single genome!

| step | threads | time |
|---|---|---|
| **BWA** | 24 | 7 |
| **samtools view** | 1 | 2 |
| **sort + index** | 1 | 3 |
| **MarkDuplicates** | 1 | 11 |
| **RealignTargets** | 24 | 1 |
| **IndelRealigner** | 24 | 6.5 |
| **BaseRecalibrator** | 24 | 1.3 |
| **PrintReads + index** | 24 | 12.3 |
| **Total** | | **44** |

*data from 24 core sandy bridge processor*

# The software is not using the cpu efficiently




*data from 24 core sandy bridge processor*

But I don’t care about performance when I can just throw more computers at this problem!

really?

# Modern CPUs are too fast

- 1,000,000,000 cycles per second
- 12+ cores per socket
- 3+ execution ports per core
- 36,000,000,000 instructions per second
- We can’t waste all this power waiting for data!

# We *need* faster memory

| | |
|---|---|
| 1 cycle on a 3 GHz processor | 1 ns |
| L1 cache reference | 0.5 ns |
| Branch mispredict | 5 ns |
| L2 cache reference | 7 ns |
| Mutex lock/unlock | 25 ns |
| Main memory reference | 100 ns |
| Compress 1K bytes with Snappy | 3,000 ns |
| Send 1Kb over 1Gbps network | 10,000 ns |
| Read 4K randomly from SSD | 150,000 ns |
| Read 1Mb sequentially from SSD | 1,000,000 ns |
| Disk seek | 10,000,000 ns |
| Read 1Mb sequentially from disk | 20,000,000 ns |
| Send packet CA->Netherlands->CA | 150,000,000 ns |

*data from sandy bridge processor*

# The java codebase has severe limitations

- More than 70% of the instructions in the current GATK pipeline are memory access — *the processor is just waiting*.

- Excessive use of strings, lists, maps and sets to handle basic data structures that are frequently used in the codebase.

- Java makes it extremely difficult to explore memory contiguity in its data structures.

- Java floating point model is incompatible with modern x86 hardware.

- Java does not offer access to the hardware for optimizations even when desired. As a result, we are forced to underutilize modern hardware.

C++ DOESN'T GIVE YOU PERFORMANCE

IT GIVES YOU CONTROL OVER PERFORMANCE

– Chandler Carruth, Google

# native GATK frameworks are open source!

Libraries / frameworks

MIT License

**gamgee**

+

GATK tool developer framework **c++**

GATK tool developer framework **java**

Toolkits

GATK License

**c++**

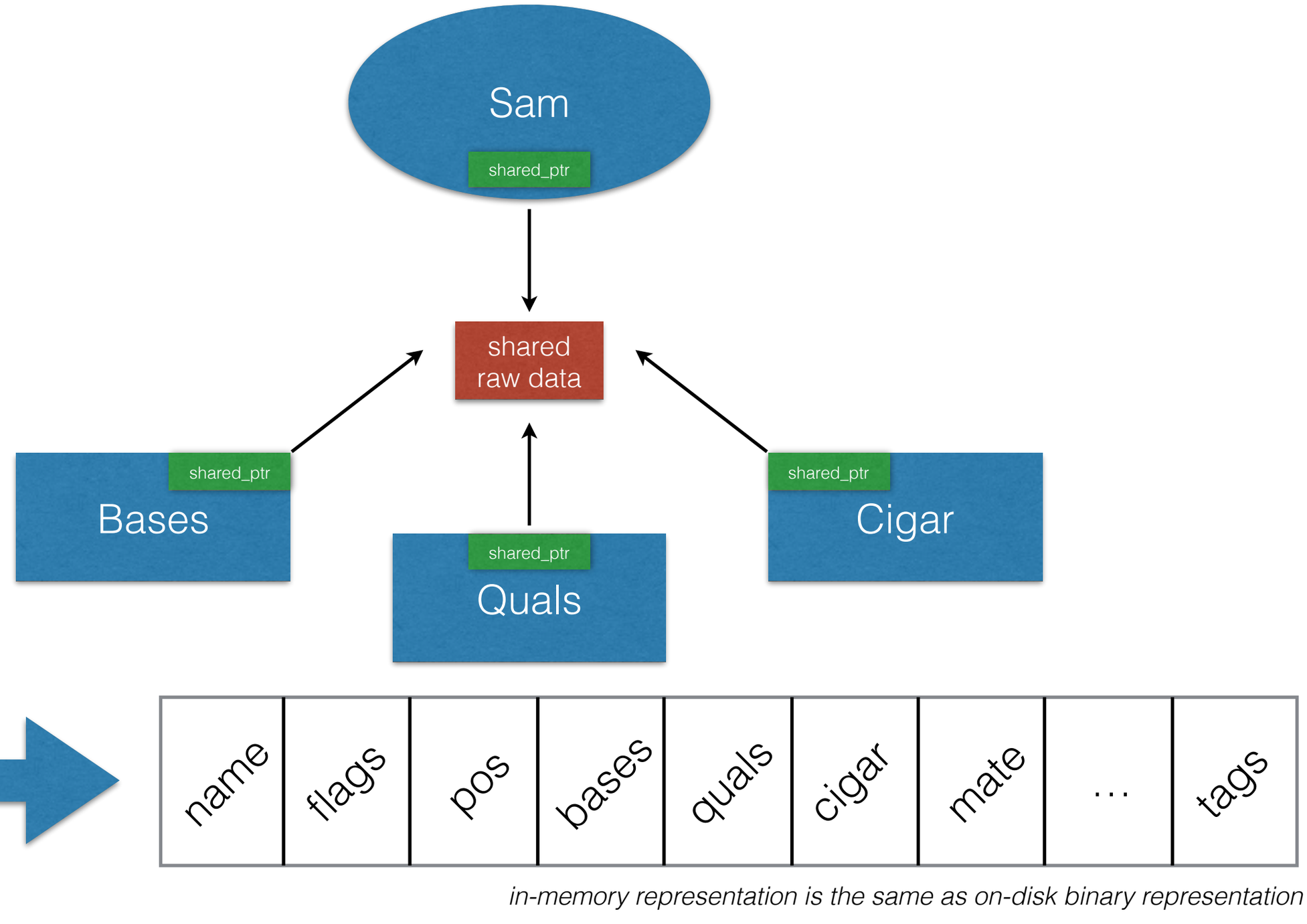
gatk

+

**java**

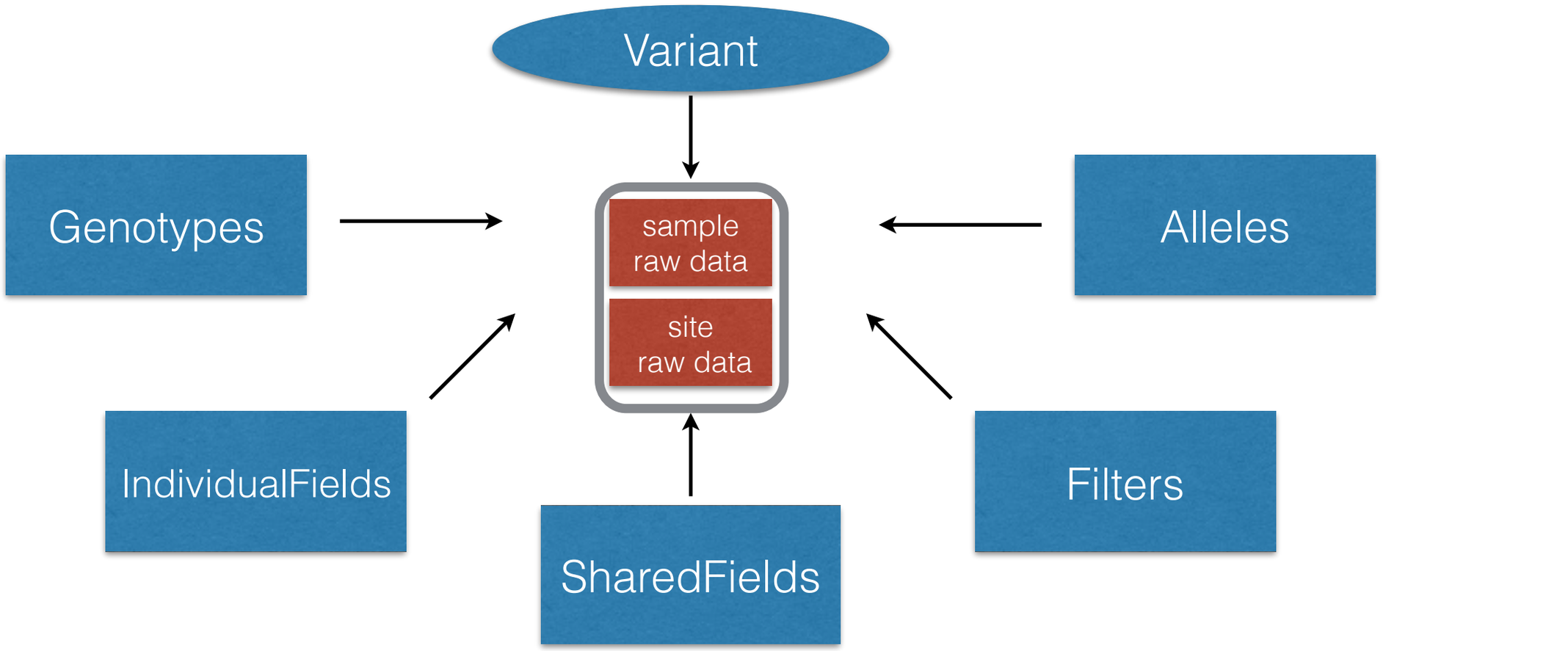
gatk

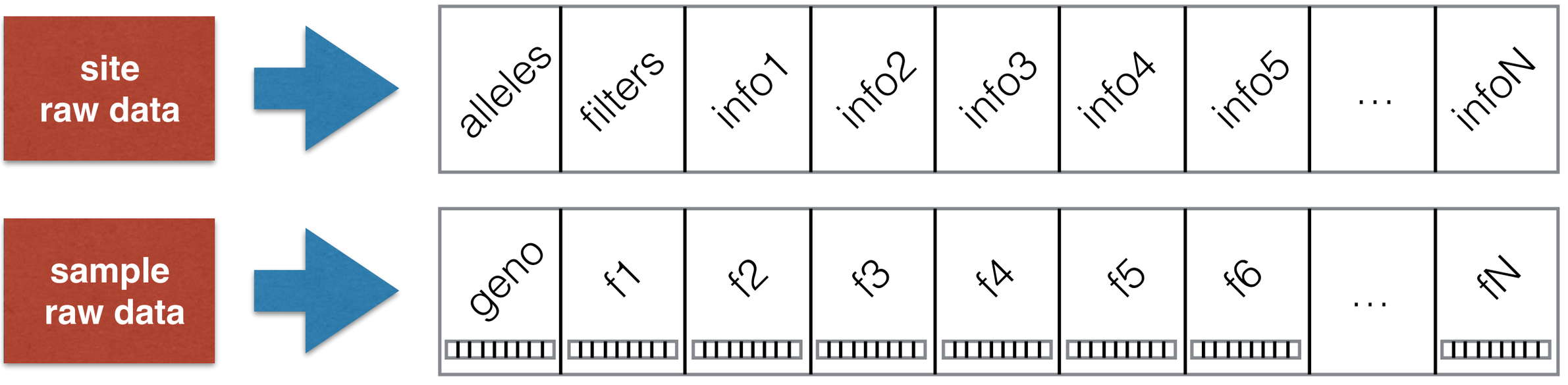
# Gamgee memory model




*in-memory representation is the same as on-disk binary representation*

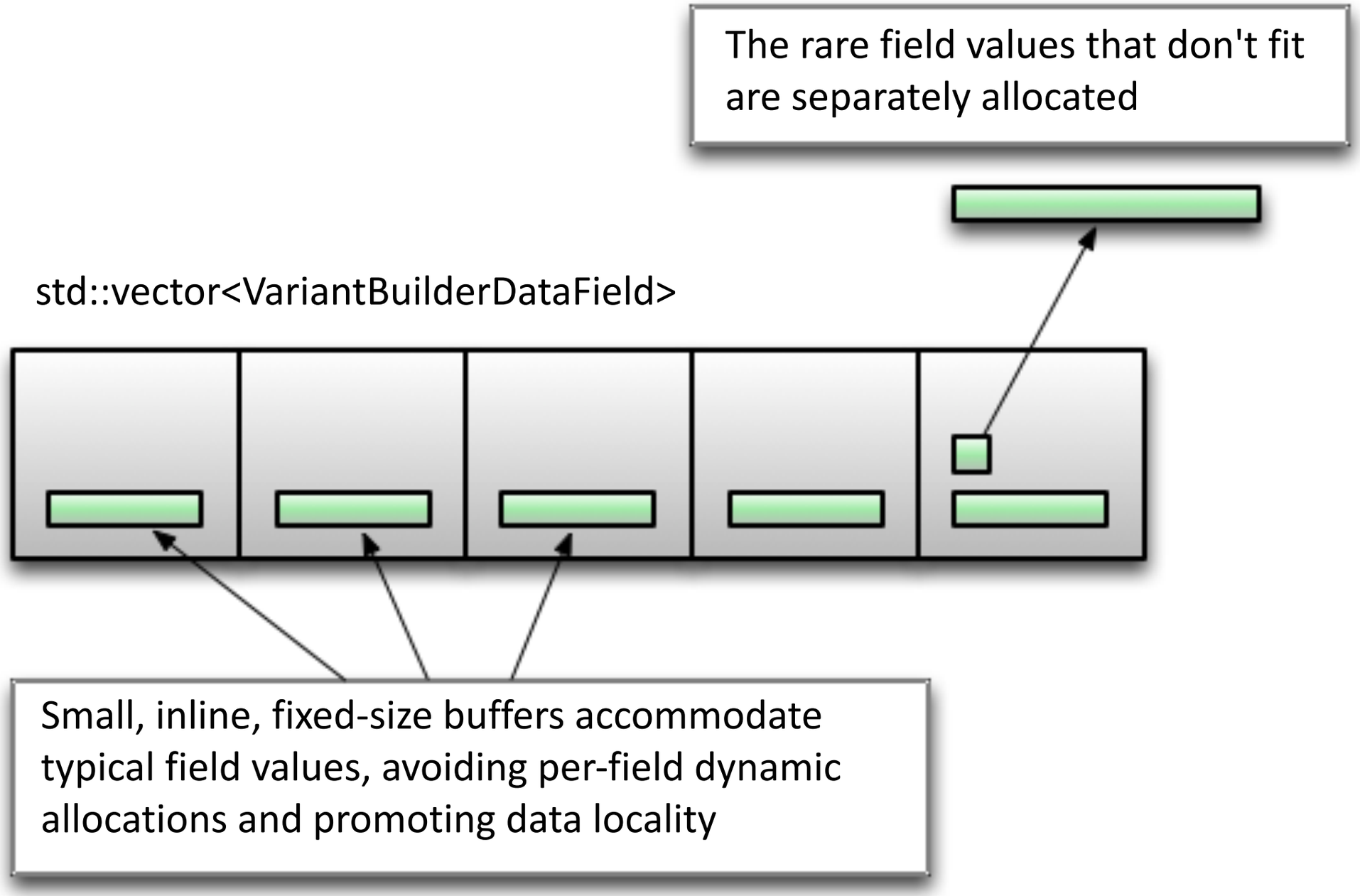
# Gamgee memory model



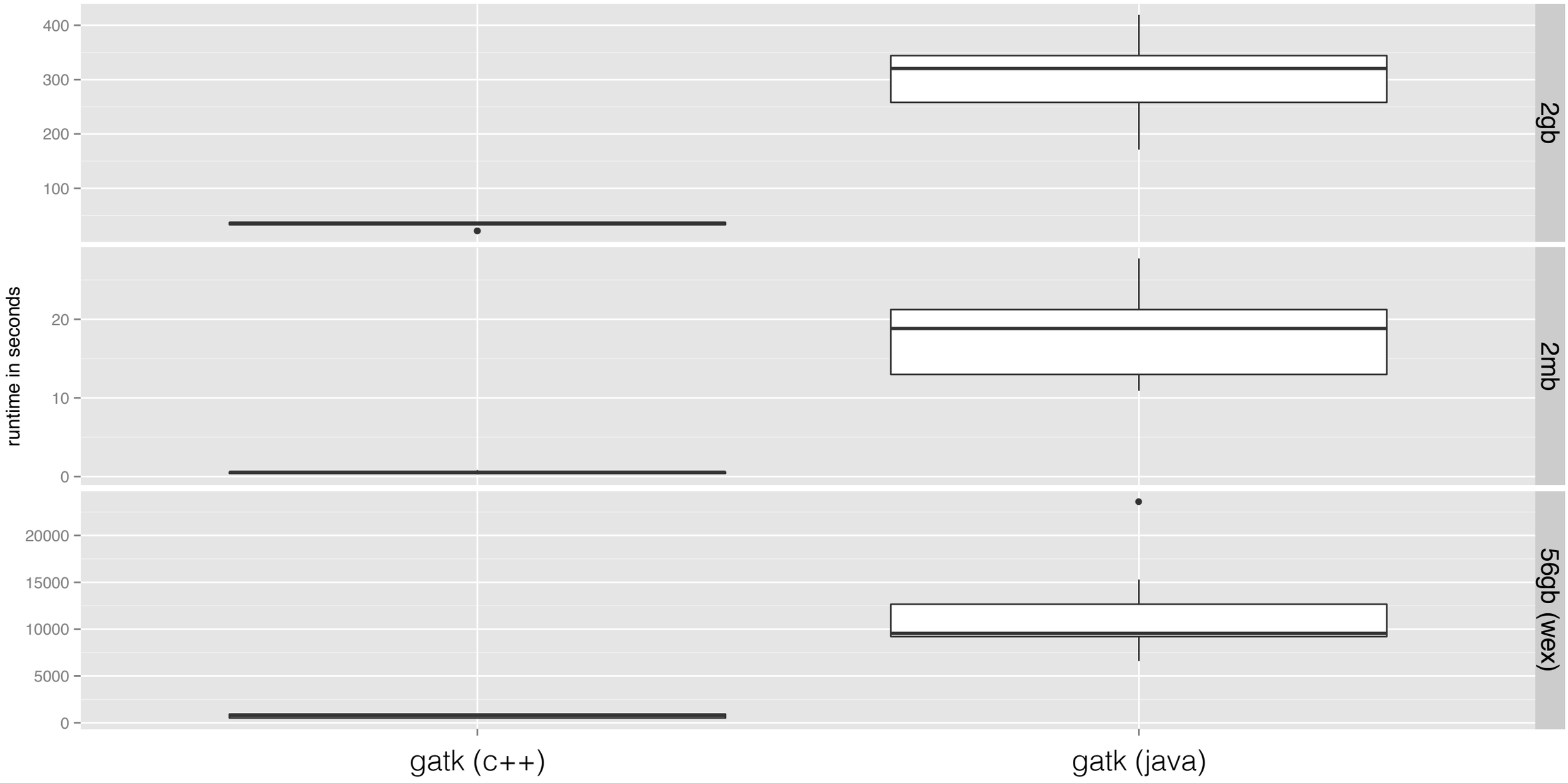

*in-memory representation is the same as on-disk binary representation*

# VariantBuilder is optimized to preserve data locality and avoid dynamic allocation as much as possible when building records




- Same idea as Short String Optimization (SSO) in std::string
- Almost impossible to achieve in Java

# Reading BAM files is 17x faster in gamgee



# Reading VCF/BCF files is much faster in gamgee

| 2GB (1KG) | GATK C++ | GATK Java |
|---|---|---|
| **Text Variant File (VCF)** | 32.71s | 137.57s |
| **Binary Variant File (BCF)** | 4.61s | 242.33s |

the new memory model makes the binary version of the file extremely fast to read and write

# Heterogeneous compute speeds up variant calling significantly

| Technology | Hardware | Runtime | Improvement |
|---|---|---|---|
| - | Java (gatk 2.8) | 10,800 | - |
| - | C++ (baseline) | 1,267 | 9x |
| FPGA | Convey Computers HC2 | 834 | 13x |
| AVX | Intel Xeon 1-core | 309 | 35x |
| GPU | NVidia GeForce GTX 670 | 288 | 38x |
| GPU | NVidia GeForce GTX 680 | 274 | 40x |
| GPU | NVidia GeForce GTX 480 | 190 | 56x |
| GPU | NVidia GeForce GTX Titan | 80 | 135x |
| GPU | NVidia Tesla K40 | 70 | 154x |
| AVX | Intel Xeon 24-core | 15 | 720x |

# MarkDuplicates is 5x faster (was 10x)

| | GATK C++ | new Picard (java) | old Picard (java) |
|---|---|---|---|
| **Exome** | 4m | 20m | 2h23m |
| **Genome** | 1h15m | 4h47m | 11h06m |

exact same implementation in Java after our C++ version was presented

# Thank you

We welcome your contribution!

Gamgee source code available:
https://github.com/broadinstitute/gamgee

start using it today!

Foghorn (*the GATK c++ engine*) will be available soon.