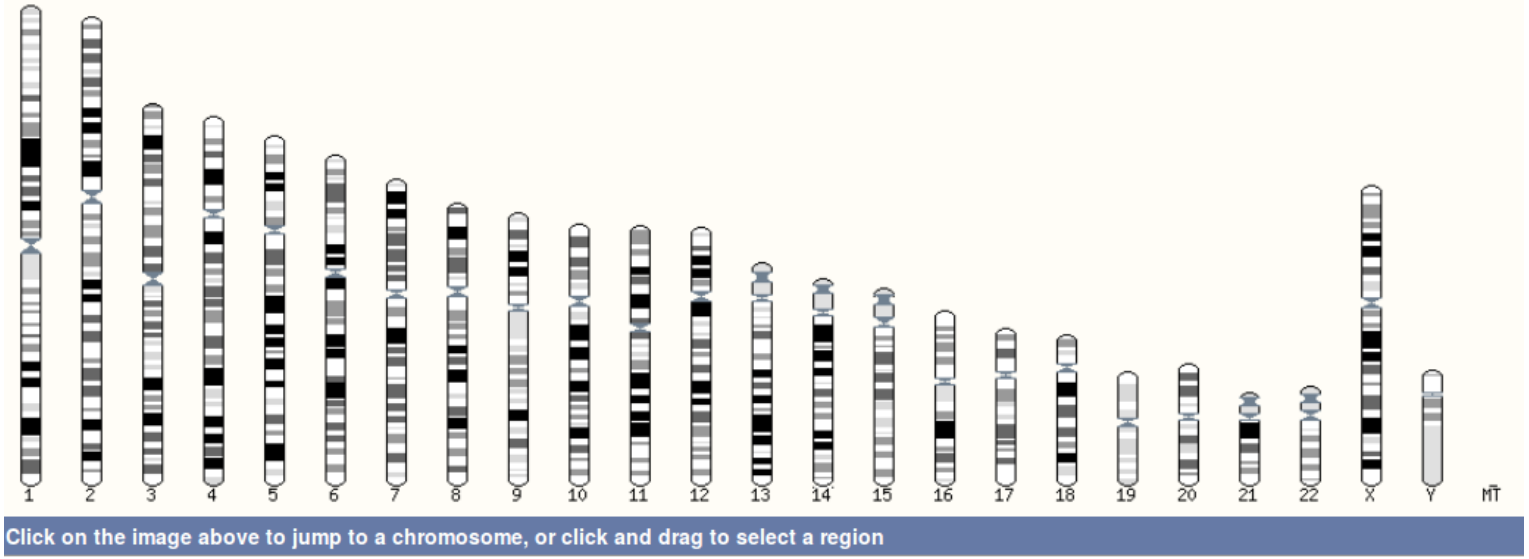
Scaling community developed variant calling analyses

Brad Chapman
Bioinformatics Core, Harvard School of Public Health
https://github.com/chapmanb/bcbio-nextgen
http://j.mp/bcbiolinks

7 August 2014

Human whole genome sequencing

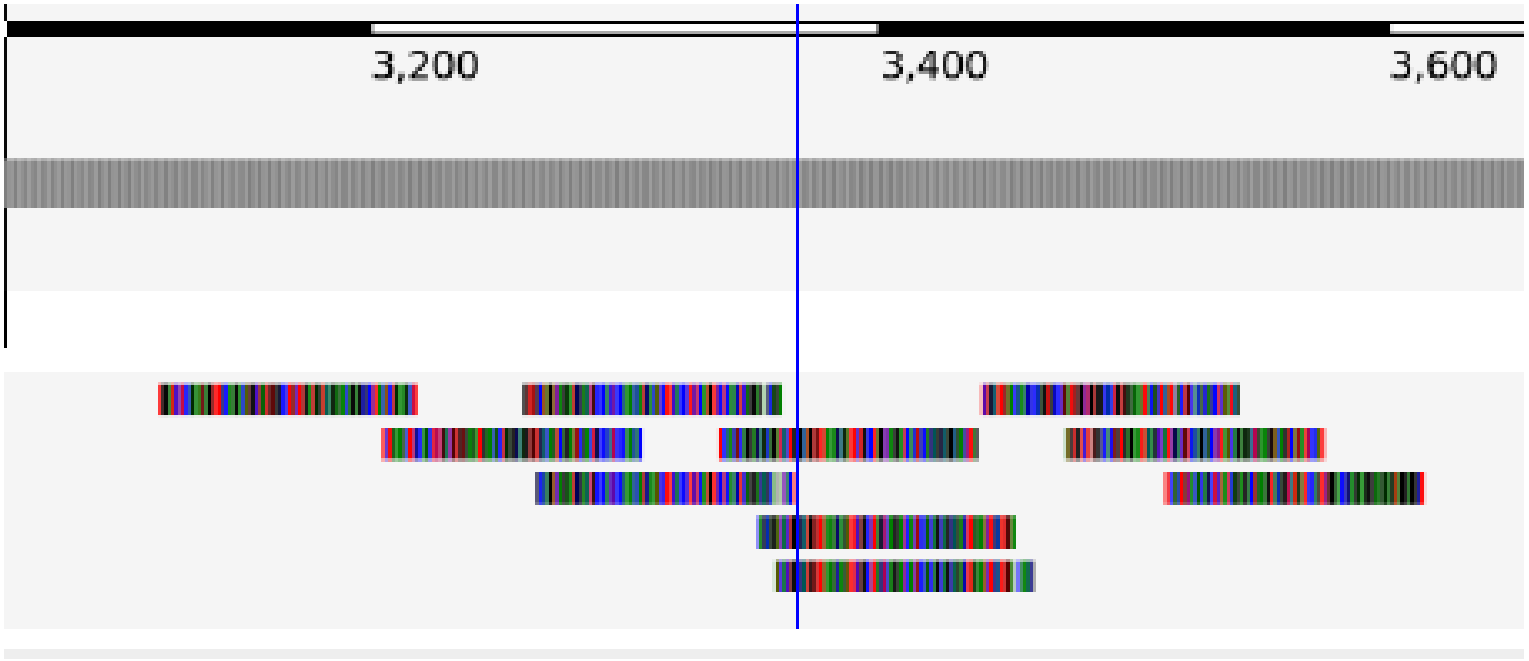


Summary

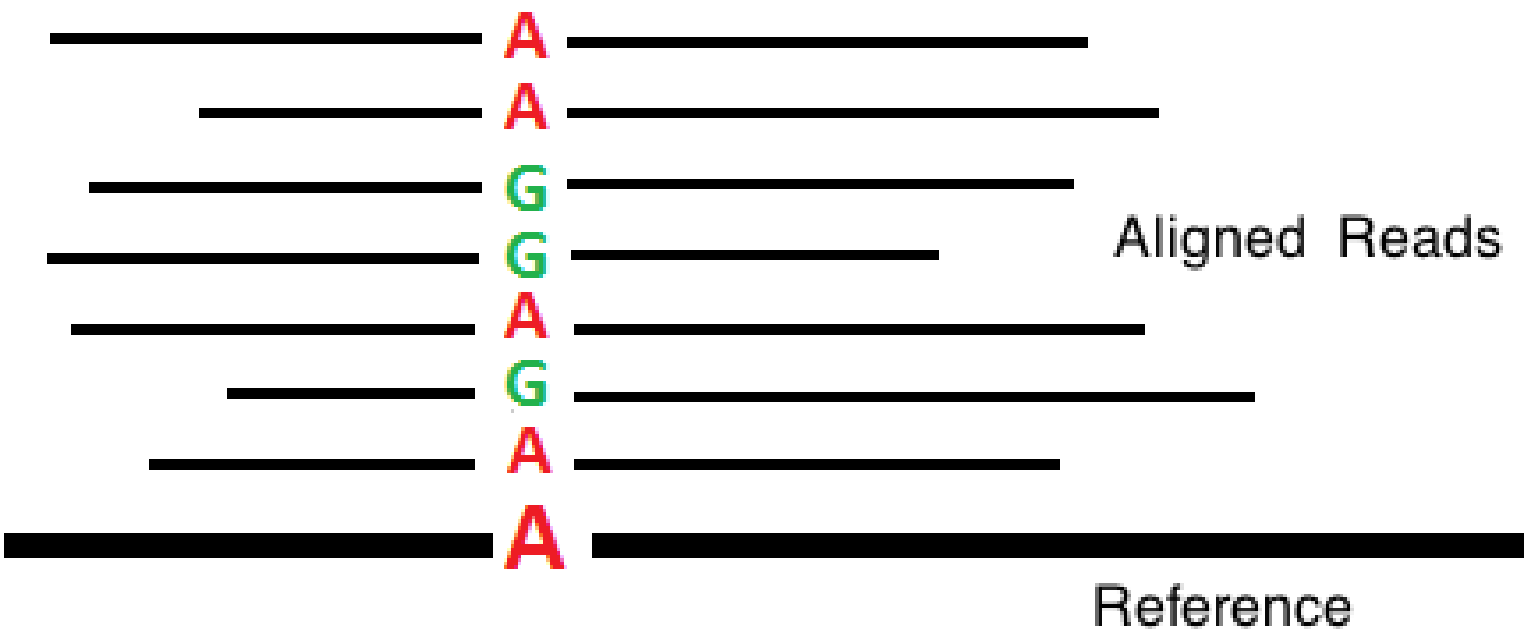
Assembly	GRCh37.p13 (Genome Reference Consortium Human Reference 37), INSDC Assembly GCA_000001405.14, Feb 2009
Database version	75.37
Base Pairs	3,326,743,047

http://ensembl.org/Homo_sapiens/Location/Genome

High throughput sequencing

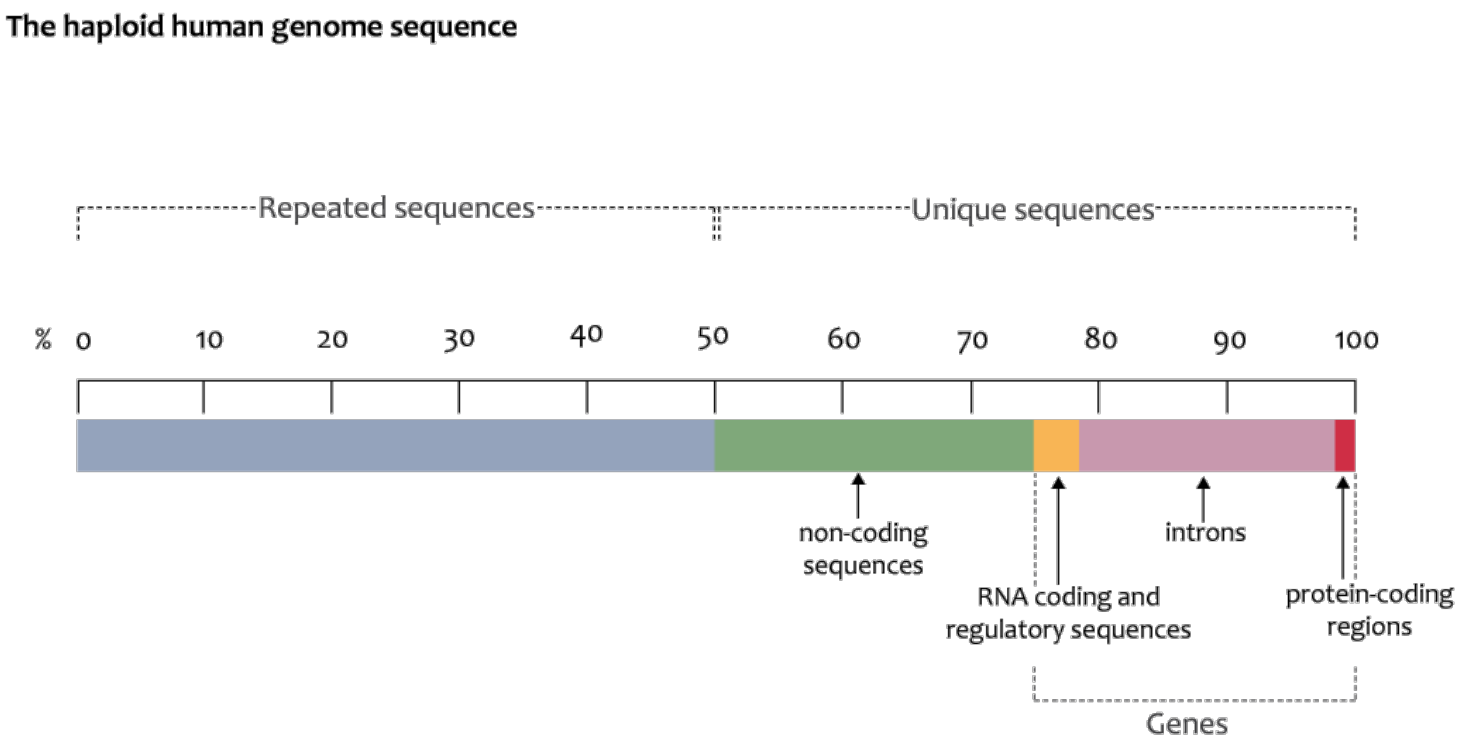


Variant calling



http://en.wikipedia.org/wiki/SNV_calling_from_NGS_data

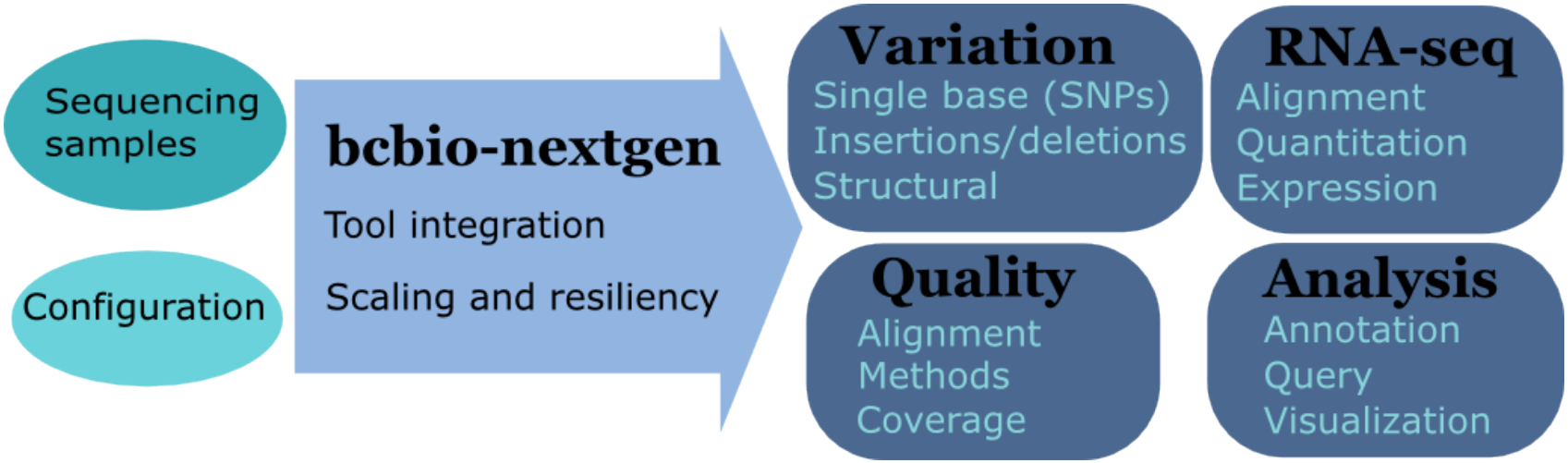
Scale: exome to whole genome



https://www.flickr.com/photos/119980645@N06/

White box software

Overview



https://github.com/chapmanb/bcbio-nextgen

Uses

- Aligners: bwa-mem, novoalign, bowtie2
- Variantion: FreeBayes, GATK, MuTecT, Scalpel, SnpEff, VEP, GEMINI, Lumpy, Delly
- RNA-seq: Tophat, STAR, cufflinks, HTSeq
- Quality control: fastqc, bamtools, RNA-SeQC
- Manipulation: bedtools, bcftools, biobambam, sambamba, samblaster, samtools, vcflib

Provides

- Community – collected set of expertise
- Tool integration
- Validation – outputs + automated evaluation
- Scaling
- Installation of tools and data

Community: documentation

bcbio-nextgen

Search docs

Docs » Getting started Edit on GitHub

Getting started

Overview

1. Create a sample configuration file for your project (substitute the example BAM and fastq names below with the full path to your sample files):

```
bcbio_nextgen.py -w template gatk-variant project1 sample1.bam sample2_1.fq sample2_2.fq
```

This uses a standard template (GATK best practice variant calling) to automate creation of a full configuration for all samples. See *Automated sample configuration* for more details on running the script, and manually edit the base template or final output file to incorporate project specific configuration. The example pipelines provide a good starting point and the *Sample information* documentation has full details on available options.

2. Run analysis, distributed across 8 local cores:

```
bcbio_nextgen.py bcbio_sample.yaml -n 8
```

https://bcbio-nextgen.readthedocs.org

Community: contribution

chapmanb / **bcbio-nextgen**

Unwatch 33 | Unstar 119 | Fork 63

Validated, scalable, community developed variant calling and RNA-seq analysis
https://bcbio-nextgen.readthedocs.org — Edit

2,717 commits | **1** branch | **16** releases | **18** contributors

branch: **master** | **bcbio-nextgen** / +

Trimming overhaul, removal of decompression of FASTQ files. ...

roryk authored 5 hours ago — latest commit 4249d607ef

bcbio	Trimming overhaul, removal of decompression of FASTQ files.	5 hours ago
config	Documentation and configuration files for running whole genome struct...	4 days ago
docs	Disambiguate and fusion fields updated in docs	2 days ago

Code
Issues 32
Pull Requests 5
Pulse
Graphs
Settings

https://github.com/chapmanb/bcbio-nextgen

Validation

Tests for implementation and methods

- Currently:
  - Family/population calling
  - RNA-seq differential expression
  - Structural variations
- Expand to:
  - Cancer tumor/normal

    http://j.mp/cancer-var-chal

Example evaluation

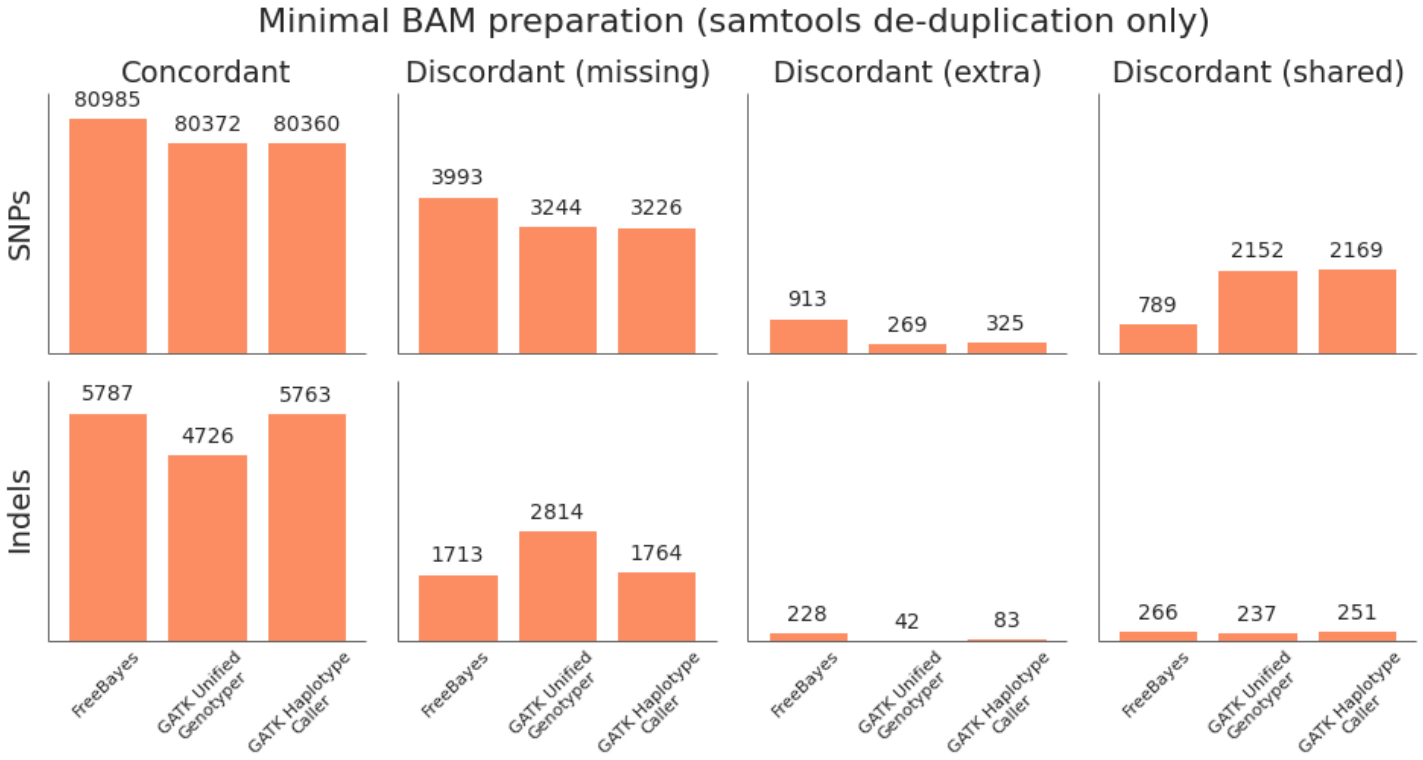
- Variant calling
  - GATK UnifiedGenotyper
  - GATK HaplotypeCaller
  - FreeBayes
- Two preparation methods
  - Full (de-duplication, recalibration, realignment)
  - Minimal (only de-duplication)

Reference materials

Genome in a Bottle Consortium

http://www.genomeinabottle.org/

Quantify quality



- Quantification details: http://j.mp/bcbioeval2

Validation enables scaling

- Little value in realignment when using haplotype aware caller
- Little value in recalibration when using high quality reads
- Streaming de-duplication approaches provide same quality without disk IO

Start point

- Initial pipeline scales with exomes
- 50 whole genomes = 3 months
- Next project: 1500 whole genomes

End point

1500 whole genome scale – 110Tb

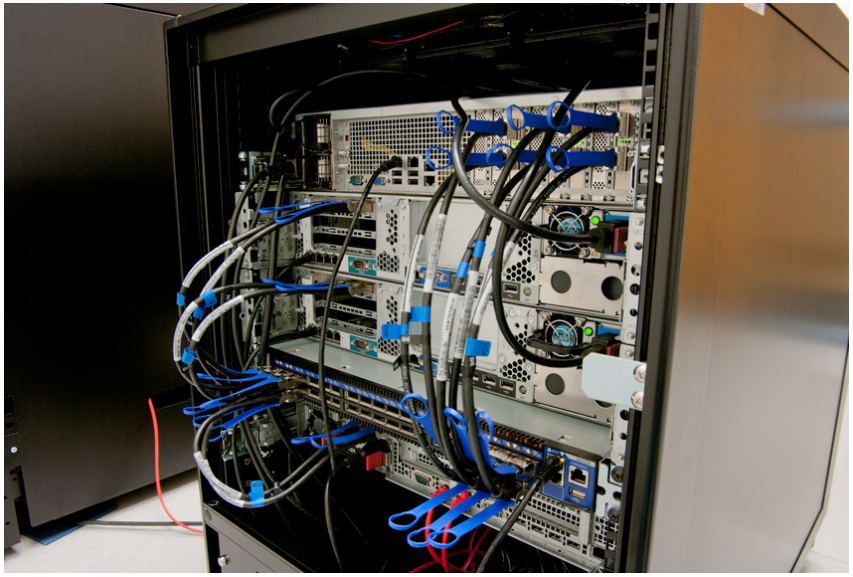
```
$ du -sh alz-p3f_2-g5/final
3.4T  alz-p3f_2-g5/final
$ ls -lhd *alz* | wc -l
31
```

How?

- Network bandwidth
- Avoid file intermediates
- Parallel alignment
- Parallel genome processing
- Better shared filesystems: Lustre

Scaling: network bandwidth

1 GigE to Infiniband



Dell Genomic Data Analysis Platform; Glen Otero

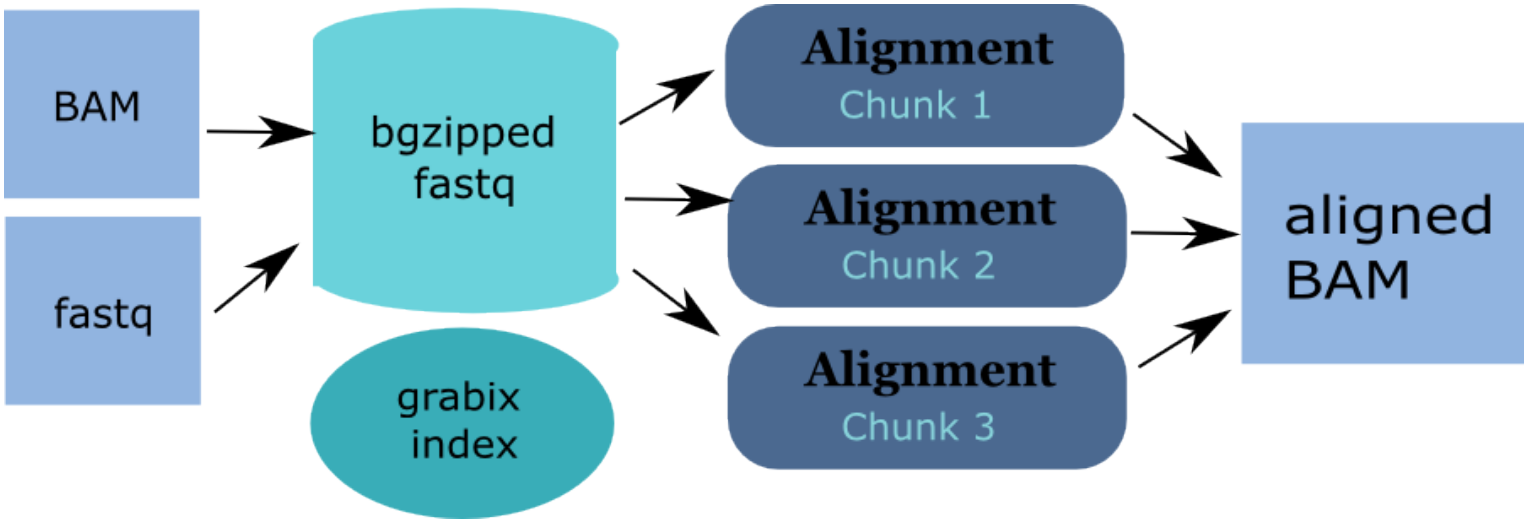
http://www.dell.com/learn/us/en/555/hpcc/

high-performance-computing-life-sciences?c=us&l=en&s=biz&cs=555

Scaling: avoid intermediates

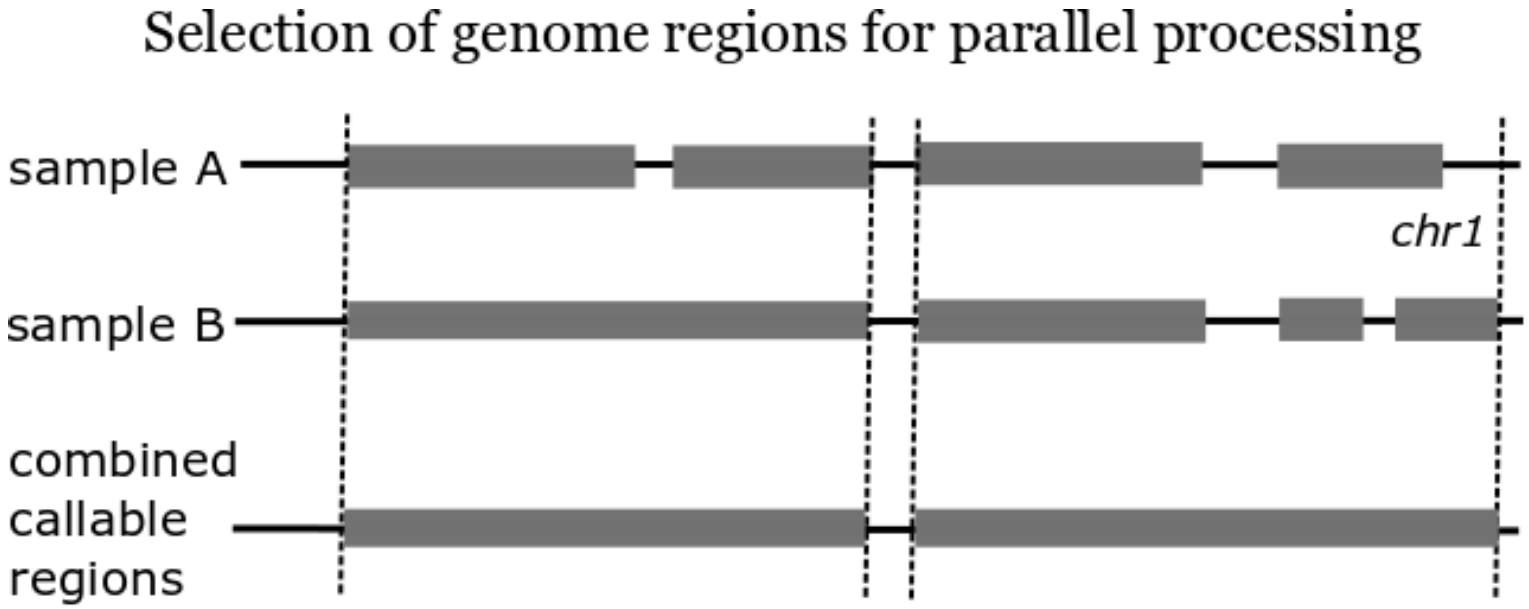
```
("{bwa} mem -M -t {num_cores} -R '{rg_info}' -v 1 "
 "  {ref_file} {fastq_file} {pair_file} "
 "| {samblaster} "
 "| {samtools} view -S -u /dev/stdin "
 "| {sambamba} sort -t {cores} -m {mem} --tmpdir {tmpdir}"
 "   -o {tx_out_file} /dev/stdin")
```

Scaling: Parallel alignment



https://github.com/arq5x/grabix

Scaling: Parallel by genome

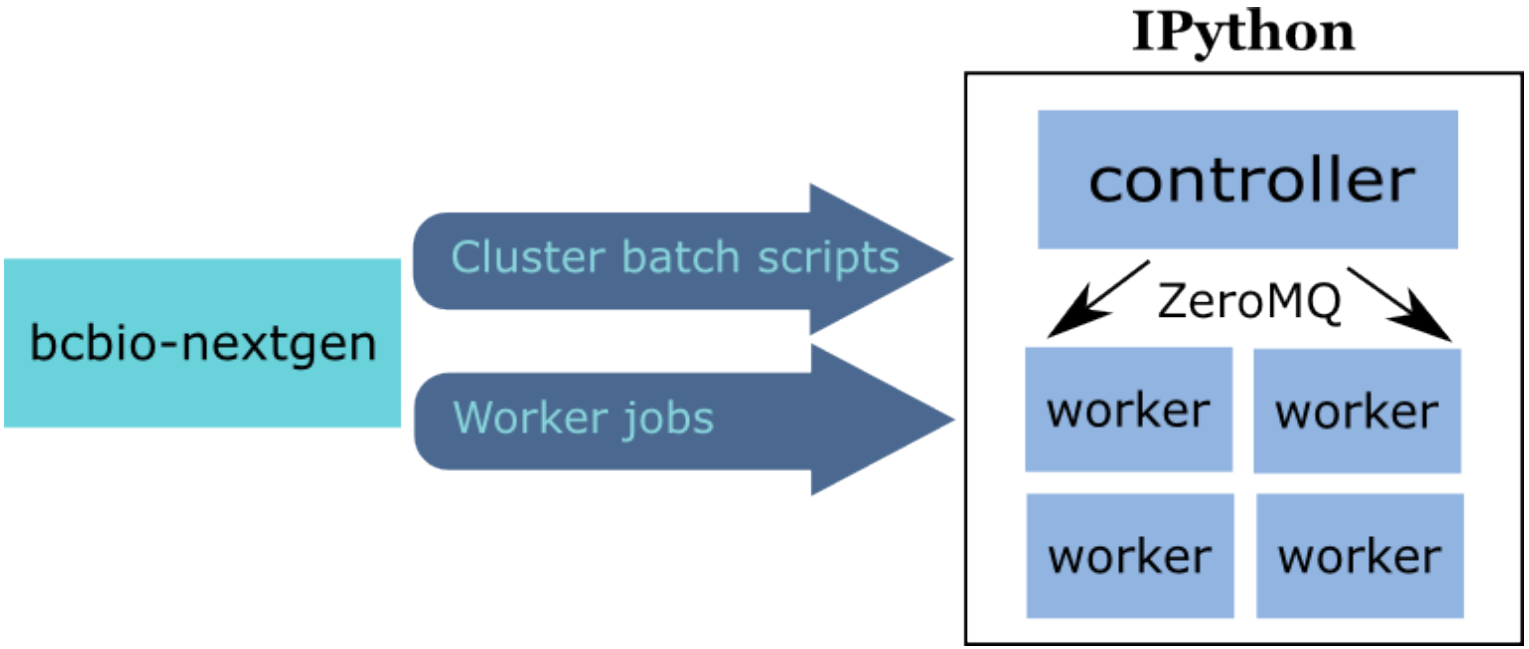


Scaling: Lustre

480 cores, 30 samples

Step	Lustre	NFS
alignment	4.5h	6.1h
alignment post-processing	7.0h	20.7h

Scaling overview



- Infrastructure details: http://j.mp/bcbioscale
- IPython: http://ipython.org/ipython-doc/dev/parallel/index.html

Current target environment

- Cluster scheduler
  - SLURM
  - Torque
  - SGE
  - LSF
- Shared filesystem
  - NFS
  - Lustre
- Local temporary disk
  - SSD

Configuration into batch scripts

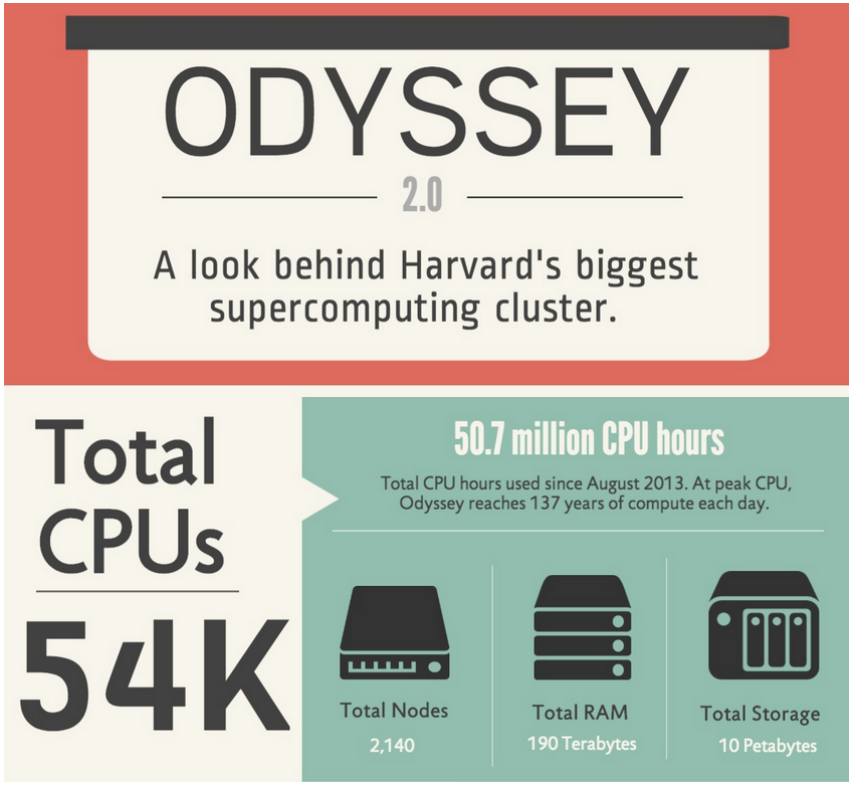
Configuration

```
bwa:
  cmd: bwa
  cores: 16
samtools:
  cores: 16
  memory: 2G
gatk:
  jvm_opts: ["-Xms750m", "-Xmx2750m"]
```

Batch file

```
#PBS -l nodes=1:ppn=16
#PBS -l mem=45260mb
```

Intel + Harvard FAS Research Computing



James Cuff, John Morrissey, Kristina Kermanshahche

https://rc.fas.harvard.edu/

Evaluation details

System

- 560 cores
- 4Gb RAM/core
- Lustre filesystem
- Infiniband network

Samples

- 75 samples
- 30x whole genome (100Gb)
- Illumina
- Family-based calling

Timing: Alignment

Step	Time	Processes
Alignment preparation	9.5 hours	BAM to fastq; bgzip; grabix index
Alignment	31 hours	bwa-mem alignment samblaster deduplication
BAM merge	5.5 hours	Merge alignment parts
Post-processing	11 hours	Calculate callable regions

Timing: Variant calling

Step	Time	Processes
Variant calling	30 hours	FreeBayes
Variant post-processing	5 hours	Combine variant files; annotate: GATK and snpEff

Timing: Analysis and QC

Step	Time	Processes
GEMINI	5 hours	Create GEMINI SQLite database
Quality Control	2.5 hours	FastQC, alignment and variant statistics

Timing: Overall

- 100 hours, ~4 days for 75 samples
- ~1 1/2 hours per sample at 560 cores
- In progress: optimize for single samples

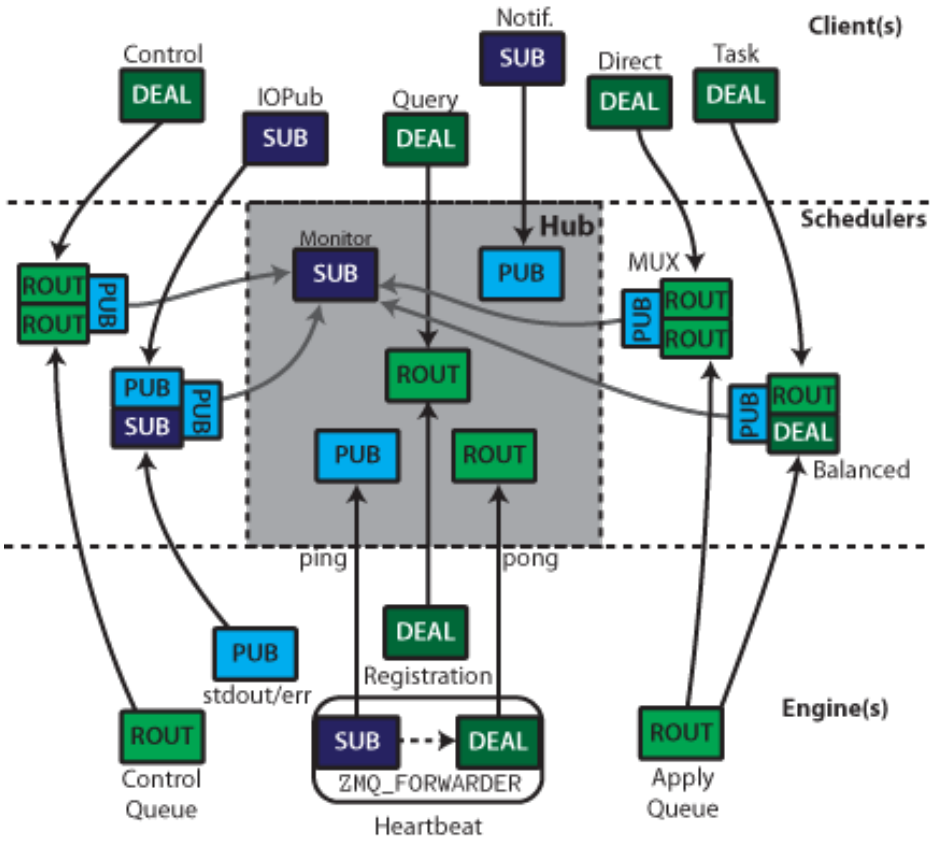
Additional scaling

- Better at profiling
- HiSeq X Ten = more genomes
- Better community support

Improved profiling

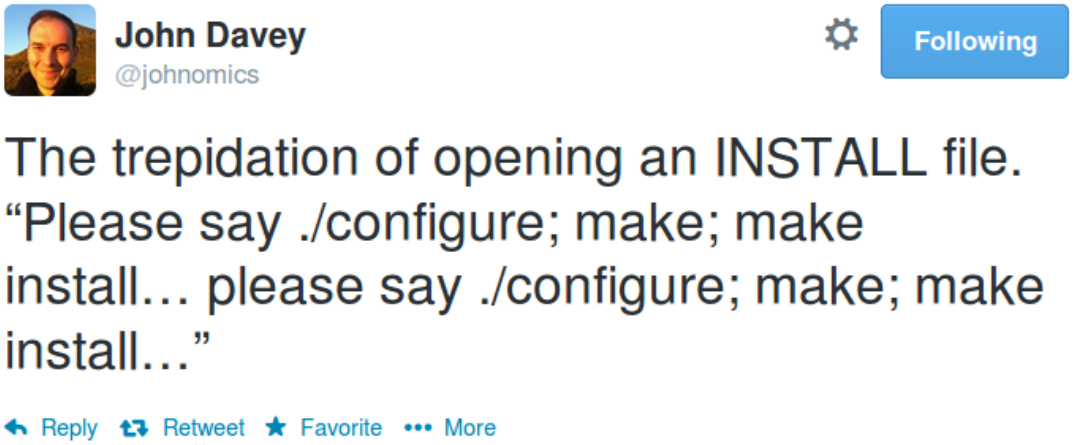
https://01.org/workflow-profiler

Improve batch size submission



http://ipython.org/ipython-doc/dev/development/parallel_messages.html

Make installation easy



Automated Install

We made it easy to install a large number of biological tools.
Good or bad idea?

Need a consistent support environment

Code 18

Issues 104

States

Closed 96

Open 8

Search all of GitHub

Installation

We've found 104 issues

Installation can fail if pypi is blocked

Opened by lbeltrame 2 days ago

Mac OS 10.9 **installation** error

Opened by alartin on Apr 13 2 comments

Update **installation**.rst

add --data to dbnsfp download

Opened by tanglingfung 26 days ago 1 comment

SHA256 mismatch for platypus-variant in **installation**

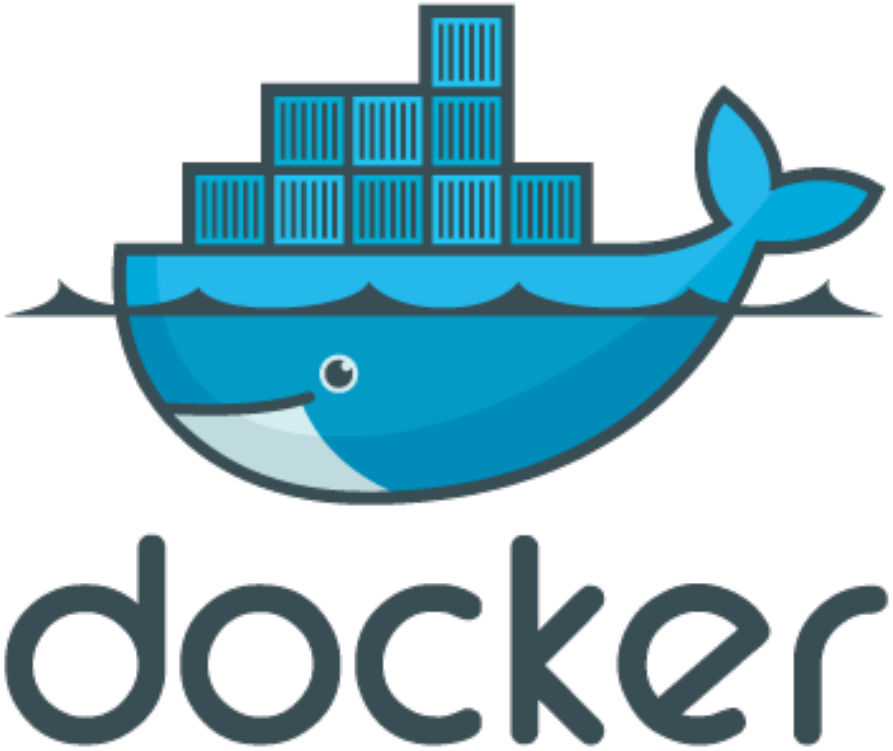
Hi, I encountered an error when installing the latest version of bcbio-nextgen on Ubuntu **installation** halted with a SHA256 mismatch error when it was installing platypus-varian

Opened by kennethban 3 days ago 2 comments

Installation in arch

Opened by kspham on Jun 12 1 comment

Docker lightweight containers



http://docker.io

Docker benefits

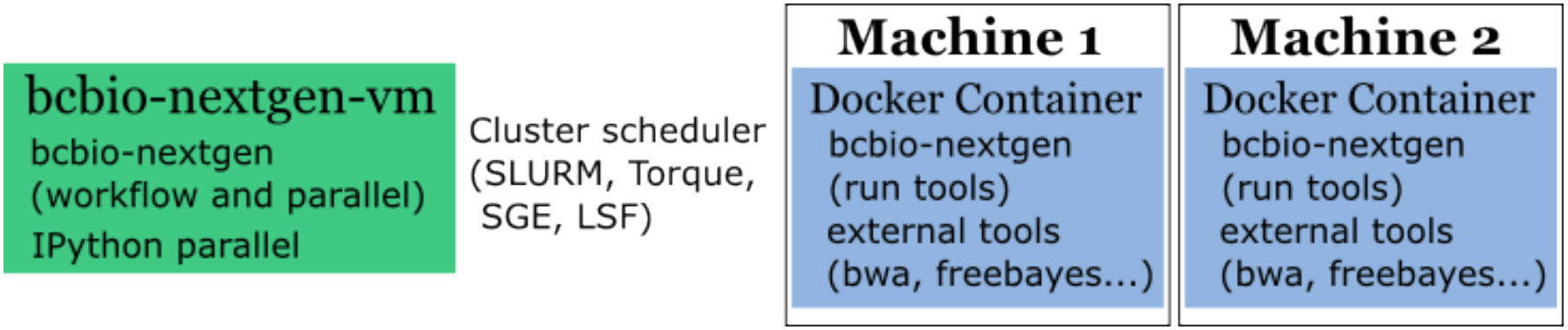
- Fully isolated
- Reproducible – store full environment with analysis (1Gb)
- Improved installation – single download + data

bcbio with Docker

- External Python wrapper
  - Installation
  - Start and run containers
  - Mount external data into containers
  - Parallelize
- All analysis tools inside Docker

https://github.com/chapmanb/bcbio-nextgen-vm
http://j.mp/bcbiodocker

Docker HPC parallelization



http://ipython.org/ipython-doc/dev/parallel/index.html
https://github.com/roryk/ipython-cluster-helper

Summary

- Community developed variant calling analyses
- Validation enables science and scaling
- Scaling from 50 to 1500 genomes
- Current batch processing timings
- To do: monitor, scale bottlenecks, improve install

https://github.com/chapmanb/bcbio-nextgen