

Community built analyses that run everywhere with bcbio

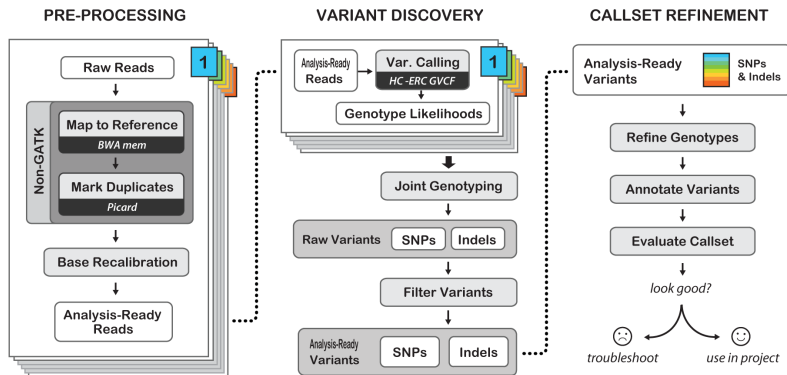
Brad Chapman
Bioinformatics Core, Harvard Chan School
<http://j.mp/bcbiolinks>

26 June 2018

- Barriers to building analysis pipelines
- bcbio: open source community development
- Common Workflow Language (CWL): assembly language for workflows
- Practical CWL with bcbio: HPC, Cloud, DNAnexus, Arvados, SevenBridges
- Personal Genome Project n=1 analysis example
- GA4GH: Automating validation and multi-platform testing

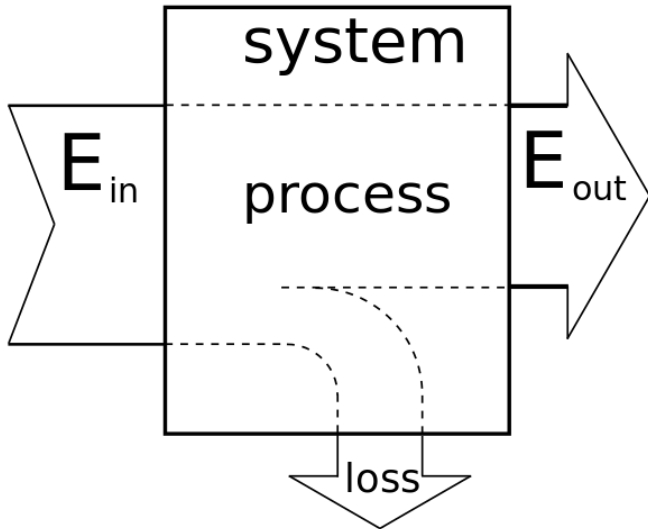
- Science = collaboration and re-use
- bcbio with interoperable workflow abstractions
- How to run bcbio analyses where you want them
- Interpreting variant calling outputs
- We can build better things together

You want to build a variant calling pipeline



Best Practices for Germline SNPs and Indels in Whole Genomes and Exomes - June 2016

<https://software.broadinstitute.org/gatk/best-practices/>



https://commons.wikimedia.org/wiki/File:Efficiency_diagram_by_Zureks.svg

Barriers to implementing yourself

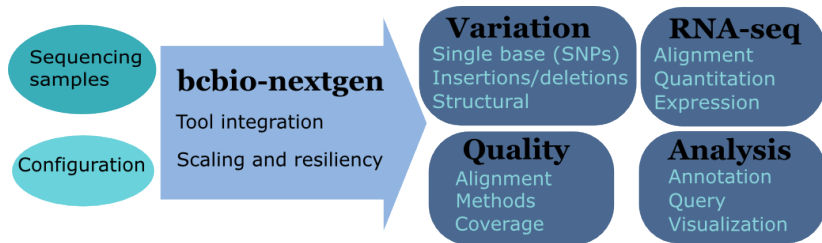
- Changing tools
- Feature support burden
- Multi-platform interoperability
- Validation

Build open source communities



<https://galaxyproject.org/events/gccbosc2018/collaboration/>

Overview



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

Supported analysis types

▢ Pipelines

▢ Germline variant calling

Basic germline calling

Population calling

Cancer variant calling

Structural variant calling

RNA-seq

single-cell RNA-seq

smallRNA-seq

ChIP-seq

<https://bcbio-nextgen.readthedocs.org/en/latest/contents/pipelines.html>

We made a pipeline – so what?

There have been a number of previous efforts to create publicly available analysis pipelines for high throughput sequencing data. Examples include Omics-Pipe, bcbio-nextgen, TREVA and NGSane. These pipelines offer a comprehensive, automated process that can analyse raw sequencing reads and produce annotated variant calls. However, the main audience for these pipelines is the research community. Consequently, there are many features required by clinical pipelines that these examples do not fully address. Other groups have focused on improving specific features of clinical pipelines. The Churchill pipeline uses specialised techniques to achieve high performance, while maintaining reproducibility and accuracy. However it is not freely available to clinical centres and it does not try to improve broader clinical aspects such as detailed quality assurance reports, robustness, reports and specialised variant filtering. The Mercury pipeline offers a comprehensive system that addresses many clinical needs: it uses an automated workflow system (Valence) to ensure robustness, abstract computational resources and simplify customisation of the pipeline. Mercury also includes detailed coverage reports provided by ExCID, and supports compliance with US privacy laws (HIPAA) when run on DNANexus, a cloud computing platform specialised for biomedical users. Mercury offers a comprehensive solution for clinical users, however it does not achieve our desired level of transparency, modularity and simplicity in the pipeline specification and design. Further, Mercury does not perform specialised variant filtering and prioritisation that is specifically tuned to the needs of clinical users.

<http://www.genomemedicine.com/content/7/1/68>

A piece of software is being sustained if people are using it, fixing it, and improving it rather than replacing it.

<http://software-carpentry.org/blog/2014/08/sustainability.html>

Complex, rapidly changing baseline functionality

Whole genome, deep coverage v1

Best Practice Variant Detection with the GATK v2

RETIRED: Best Practice Variant Detection with the GATK v3

Best Practice Variant Detection with the GATK v4, for release 2.0 [RETIRED]



Mark_DePristo Posts: 153 Administrator, GSA Member admin
July 2012 edited February 4 in [Methods and Workflows](#)

The [Best Practices](#) have been updated for GATK version 3. If you are running an older version, you should seriously consider upgrading. For more details



GATK 4.0 will be released Jan 9, 2018

Posted by [Geraldine_VdAuwers](#) on 16 Oct 2017

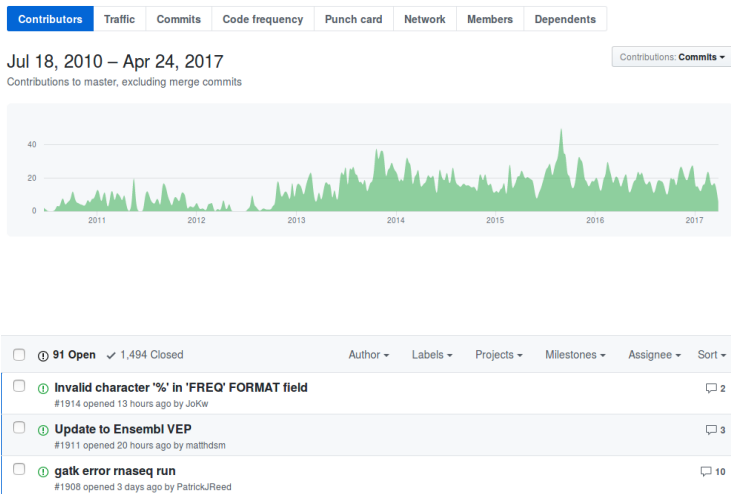
Feature support burden

Table 1: Comparison of Nextflow with other workflow management systems

Workflow	Nextflow	Galaxy	Toll	Snakemake	Bpipe
Platform^a	Groovy/JVM	Python	Python	Python	Groovy/JVM
Native task support ^b	Yes (any)	No	No	Yes (BASH only)	Yes (BASH only)
Common workflow language ^c	No	Yes	Yes	No	No
Streaming processing ^d	Yes	No	No	No	No
Dynamic branch evaluation	Yes	?	Yes	Yes	Undocumented
Code sharing integration ^e	Yes	No	No	No	No
Workflow modules ^f	No	Yes	Yes	Yes	Yes
Workflow versioning ^g	Yes	Yes	No	No	No
Automatic error takeover ^h	Yes	No	Yes	No	No
Graphical user interface ⁱ	No	Yes	No	No	No
DAG rendering ^j	Yes	Yes	Yes	Yes	Yes
Container management					
Docker support ^k	Yes	Yes	Yes	No	No
Singularity support ^l	Yes	No	No	No	No
Multi-scale containers ^m	Yes	Yes	Yes	No	No
Built-in batch schedulersⁿ					
Univa Grid Engine	Yes	Yes	Yes	Partial	Yes
PBS/Torque	Yes	Yes	No	Partial	Yes
LSF	Yes	Yes	No	Partial	Yes
SLURM	Yes	Yes	Yes	Partial	No
HTCondor	Yes	Yes	No	Partial	No
Built-in distributed cluster^o					
Apache Ignite	Yes	No	No	No	No
Apache Spark	No	No	Yes	No	No
Kubernetes	Yes	No	No	No	No
Apache Mesos	No	No	Yes	No	No
Built-in cloud^p					
AWS (Amazon Web Services)	Yes	Yes	Yes	No	No

<http://www.nature.com/nbt/journal/v35/n4/full/nbt.3820.html>

Community: sustainability and support



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

- Barriers to building analysis pipelines
- bcbio: open source community development
- **Common Workflow Language (CWL):
assembly language for workflows**
- Practical CWL with bcbio: HPC, Cloud,
DNAnexus, Arvados, SevenBridges
- Personal Genome Project n=1 analysis example
- GA4GH: Automating validation and
multi-platform testing

Infrastructure Goals

- Local machines
- Clusters: SLURM, SGE, Torque, PBS, LSF
- Clouds: Amazon, Google, Azure
- Clinical environments
- User interface for researchers
- Integrate with LIMS
- Accessible to the general public

Mike Lin Retweeted



DNAnexus, Inc. @dnanexus · 13 Jun 2013

#BigData Parking: "There's no reason **to move data** outside the #cloud. You can do **analysis** right there." ow.ly/m14Ke #genomics



Stuart Watt @morungos · 4 Mar 2014

Big upcoming change in **genomics**: **data** sets are now too large **to download** for **analysis**. **Move code to the data**, not vice versa #ibcretreat2014



Rob Schaefer @CSciBio · Jul 17

huge problem: moving **analysis** to the data, not the other way around.
[@ewanbirney](#) #ISAG2017 #BigData



Aaron Quinlan

@aaronquinlan

Following

This is the only way genomic research can scale.

Javier Quilez @jaquol

Laura Clarke: do not download the data, bring the analysis to the data
[@laurastephen](#) #gi2017

6:54 PM · 1 Nov 2017

Why do we transfer data around?

- Lots of work to setup and configure an analysis
- Hard to port scalable analysis to new environment

Many great workflow systems: Nexflow

```
#!/usr/bin/env nextflow

cheers=Channel.from "Bonjour","Ciao","Hello","Hola"

process sayHello {
  input:
  val x from cheers

  """
  echo $x world!
  """
}
```

Nextflow

Data-driven computational pipelines

Nextflow enables scalable and reproducible scientific workflows using software containers. It allows the adaptation of pipelines written in the most common scripting languages.

Its fluent DSL simplifies the implementation and the deployment of complex parallel and reactive workflows on clouds and clusters.

Find out more



Zero config



Polyglot



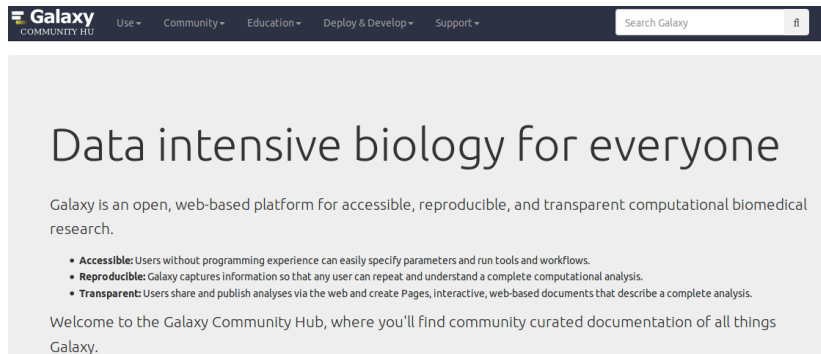
Concurrency



Scale easily

<http://nextflow.io/>

Many great workflow systems: Galaxy



The screenshot shows the top navigation bar of the Galaxy Community Hub website. It includes the Galaxy logo, a search bar, and several menu items: Use, Community, Education, Deploy & Develop, and Support. The main content area features a large heading 'Data intensive biology for everyone', a paragraph describing Galaxy as an open, web-based platform for accessible, reproducible, and transparent computational biomedical research, and a bulleted list of three key features: Accessible, Reproducible, and Transparent. Below this is a welcome message to the Galaxy Community Hub.

Galaxy
COMMUNITY HUB

Use ▾ Community ▾ Education ▾ Deploy & Develop ▾ Support ▾

Search Galaxy

Data intensive biology for everyone

Galaxy is an open, web-based platform for accessible, reproducible, and transparent computational biomedical research.

- **Accessible:** Users without programming experience can easily specify parameters and run tools and workflows.
- **Reproducible:** Galaxy captures information so that any user can repeat and understand a complete computational analysis.
- **Transparent:** Users share and publish analyses via the web and create Pages, interactive, web-based documents that describe a complete analysis.

Welcome to the Galaxy Community Hub, where you'll find community curated documentation of all things Galaxy.

<http://galaxyproject.org/>

Many great workflow systems: Snakemake

Snakemake Tutorial

This tutorial introduces the text-based workflow system [Snakemake](#). Snakemake follows the [GNU Make](#) paradigm: workflows are defined in terms of rules that define how to create output files from input files. Dependencies between the rules are determined automatically, creating a DAG (directed acyclic graph) of jobs that can be automatically parallelized.

Snakemake sets itself apart from existing text-based workflow systems in the following way. Hooking into the Python interpreter, Snakemake offers a definition language that is an extension of [Python](#) with syntax to define rules and workflow specific properties. This allows to combine the flexibility of a plain scripting language with a pythonic workflow definition. The Python language is

<https://snakemake.readthedocs.io>

But, many workflow systems

Existing Workflow systems

Michael R. Crusoe edited this page 8 hours ago · 141 revisions

Computational Data Analysis Workflow Systems

› An incomplete list

- 176. Reflow: a language and runtime for distributed, integrated data processing in the cloud
<https://github.com/grailbio/reflow>
- 177. Resolwe: an open source dataflow package for Django framework <https://github.com/genialis/resolwe>
- 178. Yahoo! Pipes (historical) https://en.wikipedia.org/wiki/Yahoo!_Pipes
- 179. Walrus <https://github.com/fjukstad/walrus>
- 180. Apache Beam <https://beam.apache.org/>
- 181. CLOSHA <https://closha.kobic.re.kr/> https://www.bioexpress.re.kr/go_tutorial <http://docplayer.net/19700397-Closha-manual-ver1-1-kobic-korean-bioinformation-center-kogun82-kribb-re-kr-2016-05-08-bioinformatics-workflow-management-system-in-bio-express.html>

<https://github.com/common-workflow-language/common-workflow-language/wiki/Existing-Workflow-systems>

We'll never agree on one system

- Advantages and disadvantages to each
- Familiarity and teaching
- Personal preference

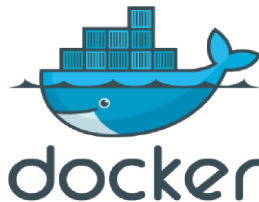
So we can't easily share workflows

- Single workflow system allows coordinated groups
- Create barrier to sharing externally
- Hard to mix and match components between workflow environments
- How can we do better?

Better abstractions = more interoperability






COMMON
WORKFLOW
LANGUAGE



<https://bcbio-nextgen.readthedocs.io/en/latest/contents/cwl.html>

Common Workflow Language (CWL)

Workflow	pipeline-se-narrow.cwl		
Sub-workflow 1	01-qc-se.cwl		
Step 1	extract.cwl	extract.py	
Step 2	count.cwl	count.py	
Step 3	fastqc.cwl	fastqc	
Sub-workflow 2	02-trim.cwl		
...			

<http://www.commonwl.org/>

<https://f1000research.com/slides/5-1617>

Workflow Description Language (WDL)



<http://openwdl.org/>

Why use a workflow abstraction?

- Integrate with multiple platforms
 - Toil – local
 - Cromwell – HPC, Cloud, local
 - Arvados
 - DNAnexus
 - Seven Bridges + Cancer Genomics Cloud
- Stop maintaining bcbio specific infrastructure
- Focus on hard biological problems

Advantages of CWL: platform resiliency

Please read

Seven Bridges will no longer develop the Open Source version of Rabix Executor. Please use the open source [CWLtool](https://github.com/rabix/bunney) instead. CWLtool is actively developed, maintained and supported by the CWL community, including Seven Bridges.

<https://github.com/rabix/bunney>

Advantages of CWL: platform resiliency



You can't spell Cromwell without CWL

Posted by [jgentry](#) on 25 May 2018

(0)

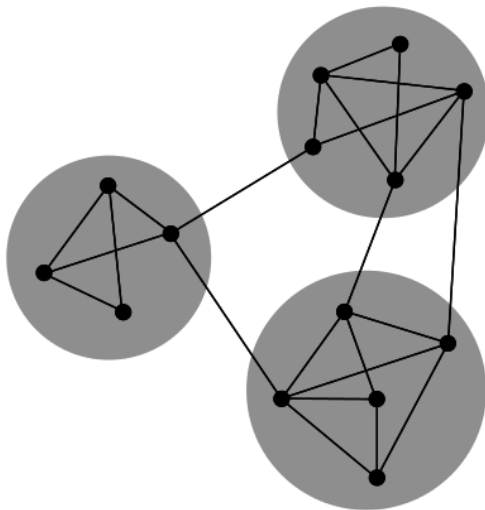
In January I [reported that Cromwell was expanding workflow language support beyond WDL](#) via a concept we call the Workflow Object Model (WOM). In that post I announced that development towards supporting the [Common Workflow Language \(CWL\)](#) was already underway. As of today, that work is complete and we have released [Cromwell version 32](#) which supports running CWL workflows.

<https://github.com/broadinstitute/cromwell/>

Unique goals with CWL

- Multiple concurrent production environments
 - HPC
 - External vendors (DNAnexus, SevenBridges, Arvados)
 - Direct on Cloud (AWS, GCE, Azure)
- Coordinated release and update process
 - Workflow
 - Tools in containers
 - Reference data

Connections



By jham3 - Own work, CC BY-SA 3.0,

<https://commons.wikimedia.org/w/index.php?curid=17125894>

- Barriers to building analysis pipelines
- bcbio: open source community development
- Common Workflow Language (CWL): assembly language for workflows
- **Practical CWL with bcbio: HPC, Cloud, DNAnexus, Arvados, SevenBridges**
- Personal Genome Project n=1 analysis example
- GA4GH: Automating validation and multi-platform testing

- Start with high level configuration file
- Generate CWL
- Run, on any infrastructure that supports CWL
 - Generated CWL
 - Docker or local bcbio installation
 - Genome data

<https://bcbio-nextgen.readthedocs.io/en/latest/contents/cwl.html>

- bcbio-like interface integrating with external tools
- Install wrapper plus supported runners

```
conda install -c conda-forge -c bioconda bcbio-nextgen-vm
```

<https://github.com/chapmanb/bcbio-nextgen-vm>

<https://bioconda.github.io/>

Practical example: Personal Genome Project

The Personal Genome Project

The Personal Genome Project, initiated in 2005, is a vision and coalition of projects across the world dedicated to creating public genome, health, and trait data. Sharing data is critical to scientific progress, but has been hampered by traditional research practices. The PGP approach is to invite willing participants to publicly share their personal data for the greater good.



<http://www.personalgenomes.org/us>

Whole genome sequencing data plus metadata

Public Profile -- huD57BBF

Real Name

James L Vick

Personal Health Records

Demographic Information

Date of Birth	1949-04-30 (69 years old)
Gender	Male
Weight	165lbs (75kg)
Height	5ft 10in (177cm)
Blood Type	O+
Race	White

<https://my.pgp-hms.org/profile/huD57BBF>

Rich set of associated data



Public data

Harvard Personal Genome Project

PGP-Harvard-huD57BBF-surveys.json

Download

(7.8 KB) PGP Harvard survey data, JSON format.

Wild Life of Our Homes

bacteria-kit-1243-graphs.png

Download

(413.2 KB) Visualization of Wild Life of Our Homes bacteria data

bacteria-kit-1243.csv.bz2

Download

(602.6 KB) Bacteria 16S-based OTU counts and taxonomic classifications

<https://www.openhumans.org/member/jameslvick/>

Template: describe your analysis

```
- files: huD57BBF.bam
  description: huD57BBF
  analysis: variant
  genome_build: hg38
  algorithm:
    aligner: bwa
    variantcaller: gatk-haplotype
    svcaller: [manta, lumpy, cnvkit]
    hlacaller: optitype
```

https://github.com/bcbio/bcbio_validation_workflows

Local filesystem environment

```
local:
  ref: biodata/collections
  inputs:
    - biodata/regions
    - biodata/pgp
resources:
  default:
    cores: 8
    memory: 3500M
    jvm_opts: [-Xms750m, -Xmx3500m]
```


Equivalent on a remote platform

```
arvados:  
  reference: su92l-4zz18-3p00f79y4p535ia  
  input: [su92l-4zz18-ihm3wrgyuwcmx1]  
resources:  
  default: {cores: 16, memory: 3500M,  
            jvm_opts: [-Xms1g, -Xmx3500m]}
```

Build Common Workflow Language description

```
bcbio_vm.py cwl --systemconfig bcbio_system-arvados.yaml \  
    pgp_sv_hla.yaml
```

Launch analysis

```
bcbio_vm.py cwlrun arvados pgp_sv_hla-workflow -- \  
--project-uuid su921-j7d0g-eoibug3nrwg8ysj
```

[https://workbench.su921.arvadosapi.com/projects/
su921-j7d0g-eoibug3nrwg8ysj](https://workbench.su921.arvadosapi.com/projects/su921-j7d0g-eoibug3nrwg8ysj)

Arvados pipeline run

postprocess_variants ▾	Complete	1h 15m / 1h 15m (1.0x)
concat_batch_variantcalls ▾	Complete	1m / 1m (1.0x)
variantcall_batch_region_3 ▾	Complete	4h 1m / 4h 1m (1.0x)
variantcall_batch_region ▾	Complete	3h 43m / 3h 43m (1.0x)
summarize_sv ▾	Complete	0m 13s / 0m 13s (1.0x)
detect_sv ▾	Complete	2h 4m / 2h 4m (1.0x)
variantcall_batch_region_2 ▾	Complete	2h 50m / 2h 50m (1.0x)
detect_sv_2 ▾	Complete	46m / 46m (1.0x)
detect_sv_3 ▾	Complete	11m / 11m (1.0x)

Generate CWL for local or HPC run

```
bcbio_vm.py cwl --systemconfig bcbio_system-local.yaml \  
    pgp_sv_hla.yaml
```

Run multicore on single machine with Toil

```
bcbio_vm.py cwlrun toil pgp_sv_hla-workflow
```

<http://toil.readthedocs.io>

Run distributed on SLURM cluster with Cromwell

```
bcbio_vm.py cwlrun cromwell pgp_sv_hla-workflow \  
  --no-container \  
  -q your_queue -s slurm -r timelimit=0-12:00
```

<http://cromwell.readthedocs.io>

Run on DNAnexus platform

```
dnanexus:  
  project: PGP  
  ref:  
    project: bcbio_resources  
    folder: /reference_genomes  
  inputs:  
    - /data/input  
resources:  
  default:  
    cores: 8  
    memory: 3500M  
    jvm_opts: [-Xms750m, -Xmx3500m]
```

<https://platform.dnanexus.com>

DNAnexus: upload configuration

```
PNAME=pgp_sv_hla
TEMPLATE=svcall
PROJECT=PGP

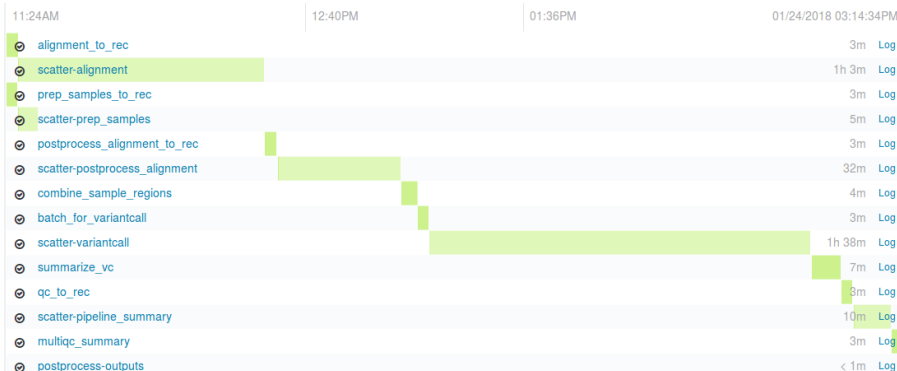
dx select $PROJECT
dx mkdir -p $PNAME
for F in $TEMPLATE.yaml $PNAME.csv bcbio_system.yaml
do
    dx rm -a /$PNAME/$F || true
    dx upload --path /$PNAME/ $F
done
```

DNAexus: run bcbio CWL applet

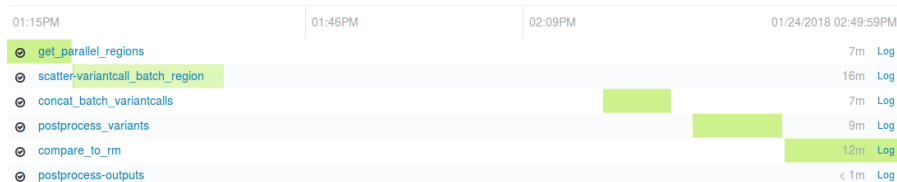
```
dx run bcbio_resources:/applets/bcbio-run-workflow \  
-iyaml_template=/$PNAME/$TEMPLATE.yaml \  
-isample_spec=/$PNAME/$PNAME.csv \  
-isystem_configuration=/$PNAME/bcbio_system.yaml \  
-ioutput_folder=/$PNAME/dx-cwl-run
```

<https://github.com/bcbio/bcbio-dnanexus-wrapper>

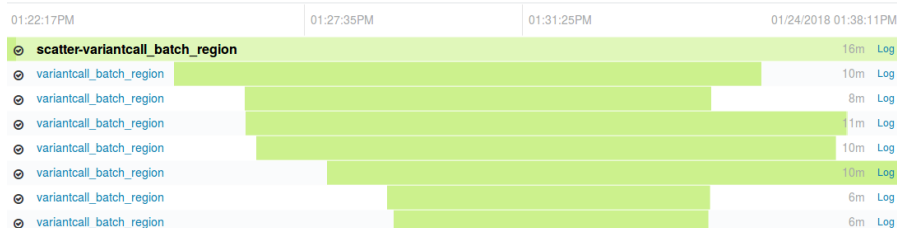
DNAnexus monitoring: align, variant call, QC



Subworkflow parallelization: per sample or batch



Variant calling parallelization: per region



Region problem: long tail jobs

03:06PM	04:41PM	05:50PM	01/09/2018 07:50:59PM
⊙ scatter-variantcall_batch_region			4h 44m Log
⊙ variantcall_batch_region			23m Log
⊙ variantcall_batch_region			24m Log
⊙ variantcall_batch_region			23m Log
⊙ variantcall_batch_region			22m Log
⊙ variantcall_batch_region			4h 40m Log
⊙ variantcall_batch_region			10m Log
⊙ variantcall_batch_region			5m Log

Region improvement: multicore Spark parallelization

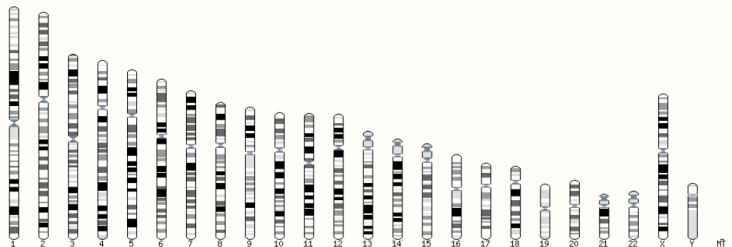
02:01AM	02:28AM	02:47AM	01/19/2018 03:20:48AM
⊙ scatter-variantcall_batch_region			1h 19m Log
⊙ variantcall_batch_region			20m Log
⊙ variantcall_batch_region			18m Log
⊙ variantcall_batch_region			22m Log
⊙ variantcall_batch_region			19m Log
⊙ variantcall_batch_region			1h 16m Log
⊙ variantcall_batch_region			8m Log
⊙ variantcall_batch_region			5m Log

- Barriers to building analysis pipelines
- bcbio: open source community development
- Common Workflow Language (CWL): assembly language for workflows
- Practical CWL with bcbio: HPC, Cloud, DNAnexus, Arvados, SevenBridges
- **Personal Genome Project n=1 analysis example**
- GA4GH: Automating validation and multi-platform testing

Looking at data – interpreting the results

- Overview of variant calling
- Example of bcbio outputs in PGP data, n=1 analysis
- Small variants
- HLA calls
- Structural variants

Human whole genome sequencing



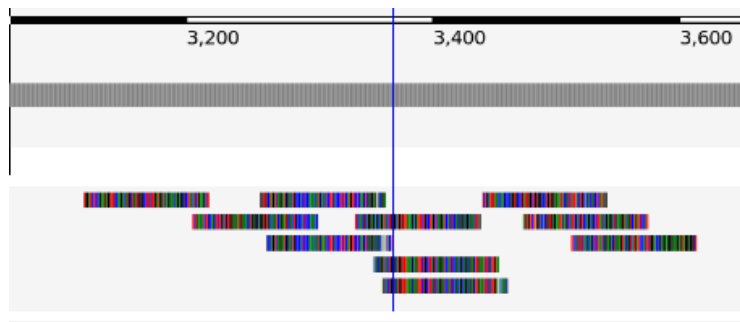
Click on the image above to jump to a chromosome, or click and drag to select a region

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

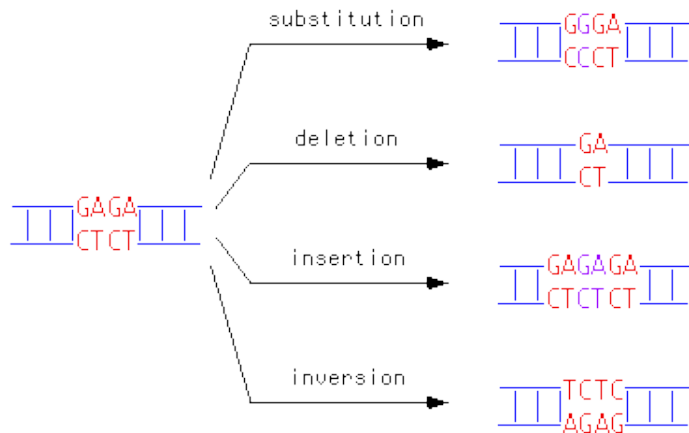


Variant calling



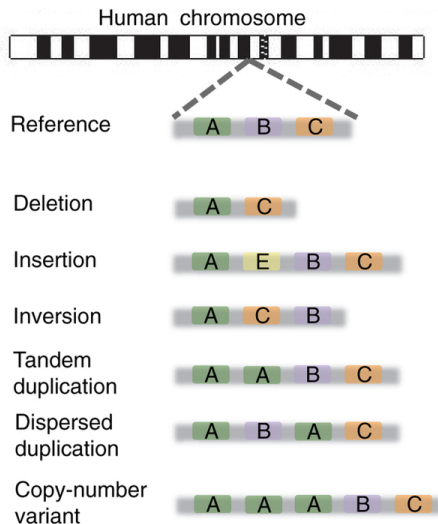
http://en.wikipedia.org/wiki/SNV_calling_from_NGS_data

SNPs and Indels



<http://carolguze.com/text/442-2-mutations.shtml>

Structural variations



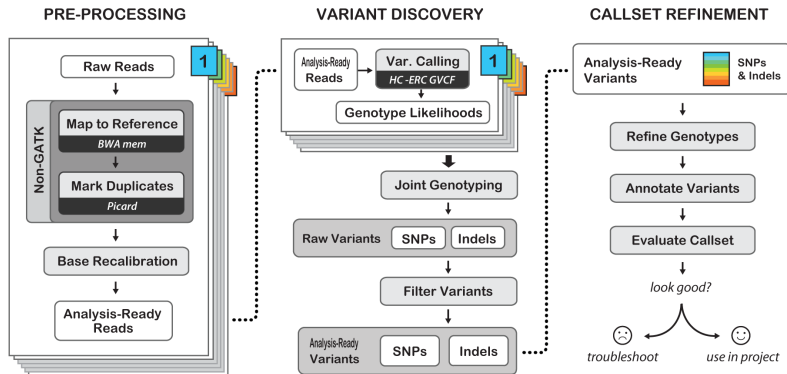
Genome Analysis Toolkit (GATK)

The Genome Analysis Toolkit or GATK is a software package developed at the Broad Institute to analyze high-throughput sequencing data. The toolkit offers a wide variety of tools, with a primary focus on variant discovery and genotyping as well as strong emphasis on data quality assurance. Its robust architecture, powerful processing engine and high-performance computing features make it capable of taking on projects of any size.



<https://www.broadinstitute.org/gatk/>

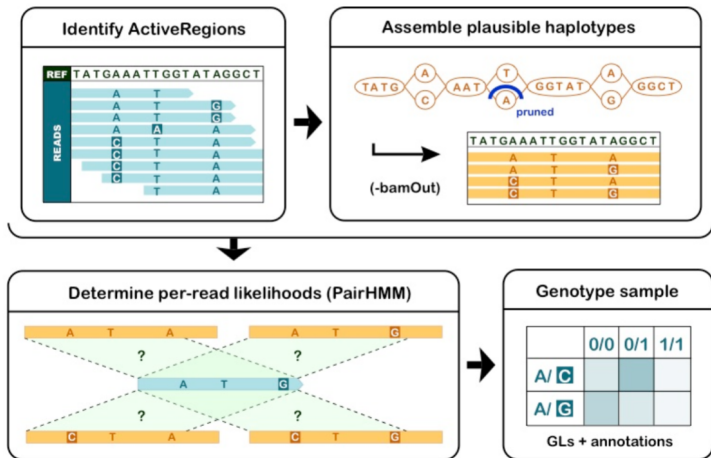
GATK Best Practices



Best Practices for Germline SNPs and Indels in Whole Genomes and Exomes - June 2016

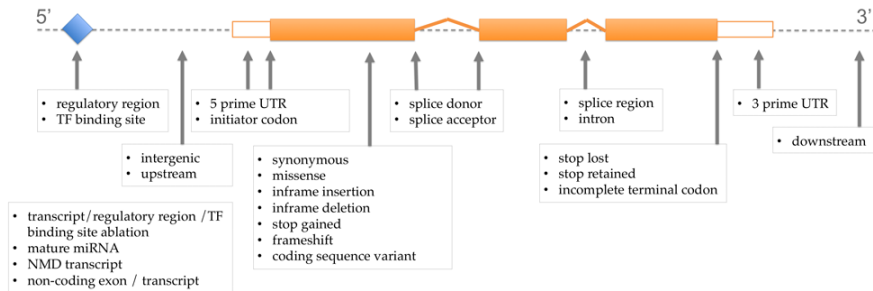
<https://software.broadinstitute.org/gatk/best-practices/>

HaplotypeCaller



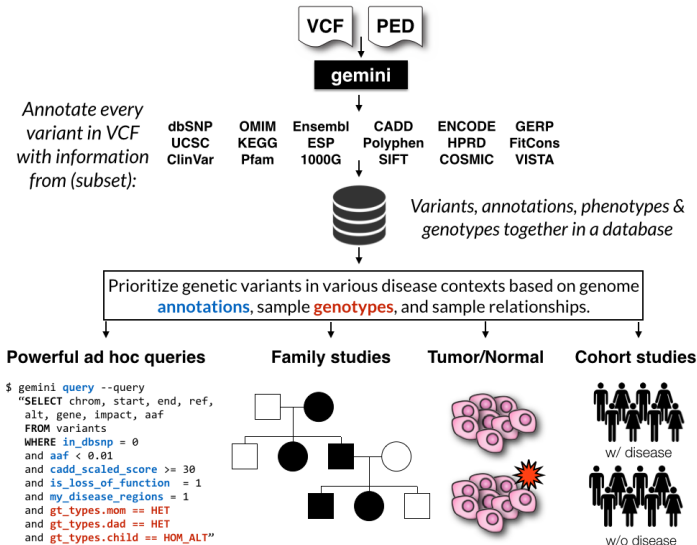
<http://gatkforums.broadinstitute.org/discussion/5464/workshop-presentations-2015-uk-4-20-24>

Effects prediction



http://www.ensembl.org/info/genome/variation/predicted_data.html

Annotation and analysis – GEMINI



<https://github.com/arq5x/gemini>

VCF – overview

VCF header

```
##fileformat=VCFv4.0
##fileDate=20100707
##source=VCFTools
##reference=NCBI36
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality (phred score)">
##FORMAT=<ID=GL,Number=3,Type=Float,Description="Likelihoods for RR,RA,AA genotypes (R=ref,A=alt)">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
```

Body

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SAMPLE1	SAMPLE2
1	1	.	ACG	A,AT	.	PASS	.	GT:DP	1/2:13	0/0:29
1	2	rs1	C	T,CT	.	PASS	H2;AA=T	GT:GQ	0/1:100	2/2:70
1	5	.	A	G	.	PASS	.	GT:GQ	1/0:77	1/1:95
1	100	.	T		.	PASS	SVTYPE=DEL;END=300	GT:GQ:DP	1/1:12:3	0/0:20

Annotations:

- Mandatory header lines:** Indicated by a red arrow pointing to the first line of the header.
- Optional header lines:** Indicated by a grey arrow pointing to the remaining header lines.
- Deletion:** Indicated by a blue arrow pointing to the ALT field of the fourth row.
- SNP:** Indicated by a blue arrow pointing to the ALT field of the second row.
- Large SV:** Indicated by a blue arrow pointing to the ALT field of the fourth row.
- Insertion:** Indicated by a blue arrow pointing to the ALT field of the second row.
- Other event:** Indicated by a blue arrow pointing to the ALT field of the second row.
- Phased data:** Indicated by a blue arrow pointing to the FORMAT field of the fourth row.

<http://vcftools.sourceforge.net/VCF-poster.pdf>

VCF – representations

Types of variants

SNPs

Alignment	VCF representation
ACGT	POS REF ALT
ATGT	2 C T

Insertions

Alignment	VCF representation
AC-GT	POS REF ALT
ACTGT	2 C CT

Deletions

Alignment	VCF representation
ACGT	POS REF ALT
A--T	1 ACG A

Complex events

Alignment	VCF representation
ACGT	POS REF ALT
A-TT	1 ACG AT

Large structural variants

VCF representation

POS	REF	ALT	INFO
100	T		SVTYPE=DEL;END=300

<http://vcftools.sourceforge.net/VCF-poster.pdf>

- Step by step guide from Broad

<https://www.broadinstitute.org/gatk/guide/article?id=1268>

- Specification

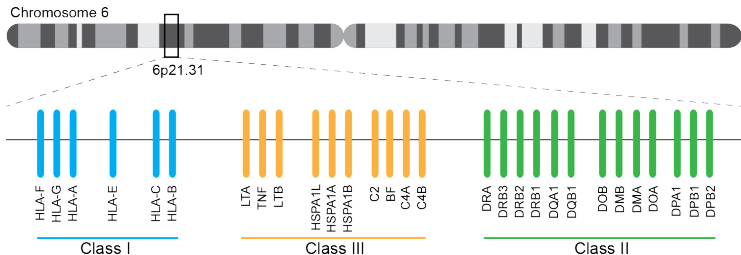
<http://samtools.github.io/hts-specs/>

ApoE analysis

```
$ tabix huD57BBF-gatk-haplotype.vcf.gz  
chr19:44908684-44908684  
chr19    44908684    rs429358    T    C  
1116.80    PASS  
ANN=C|missense_variant|MODERATE|APOE|c.388T>C|p.Cys130Arg  
GT:AD:DP:GQ:MMQ:PL    1/1:0,26:26:78:60:1145,78,0  
$ tabix huD57BBF-gatk-haplotype.vcf.gz  
chr19:44908822-44908822
```

<http://bit.ly/pgp-analysis>

Major histocompatibility complex (MHC) – HLAs



<http://www.ebi.ac.uk/ipd/imgt/hla/>

<http://sciscogenetics.com/technology/human-leukocyte-antigen-complex/>

HLA typing

- 1000 genomes: build 38 + IMGT/HLA-3.18.0
- bwa mem extracts HLA reads
- Map reads only to HLA sequences
- OptiType: Call HLA types

<https://github.com/lh3/bwa/blob/master/README-alt.md#hla-typing>

<https://github.com/FRED-2/OptiType>

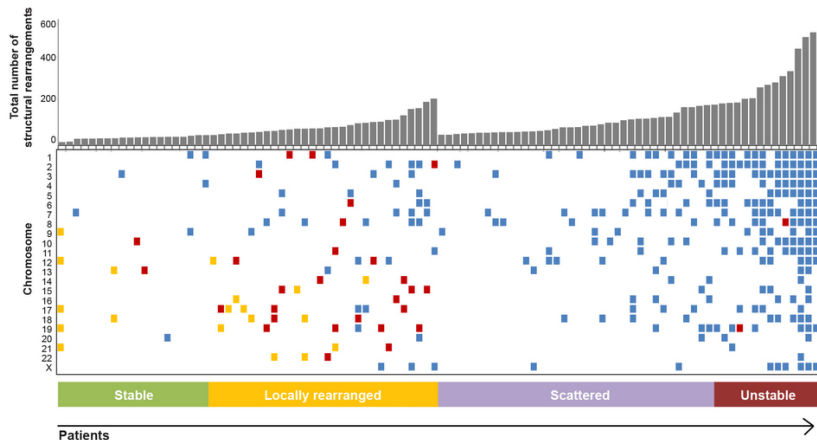
HLA outputs

HLA-A*11:01;HLA-A*24:02

HLA-B*27:05;HLA-B*55:01

HLA-C*07:02;HLA-C*07:02

Structural variants critical – pancreatic cancer example



<http://www.nature.com/nature/journal/v518/n7540/full/nature14169.html>

Tools used

- Manta: <https://github.com/Illumina/manta>
Split and paired end reads
- Lumpy: <https://github.com/arq5x/lumpy-sv>
Split and paired ends reads
- CNVkit: <https://github.com/etal/cnvkit>
Read depth based

Example deletion call – 3 callers

```
chr19    50827242          MantaDEL:67020:0:1:0:0:0
T    <DEL>    658.0 PASS
END=50830636;SVTYPE=DEL;SVLEN=-3394;
ANN=<DEL>|bidirectional_gene_fusion|HIGH|AC011523.2&KLK15|
ENSG00000267968&ENSG00000174562|gene_variant|
GT:FT:GQ:PL:PR:SR          0/1:PASS:504:708,0,501:18,16:23,12
```

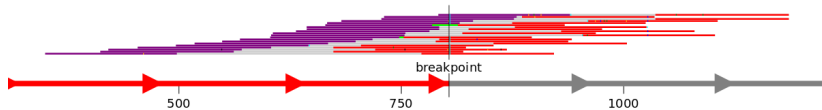
Viewing deletion – svviz

Deletion::chr19:50,827,241-50,830,635(3394)

Sample	Alt	Ref	Amb
huD57BBF-sort	20	191	146
Total	20	191	146

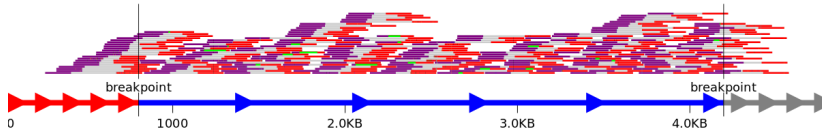
Alternate Allele

huD57BBF-sort



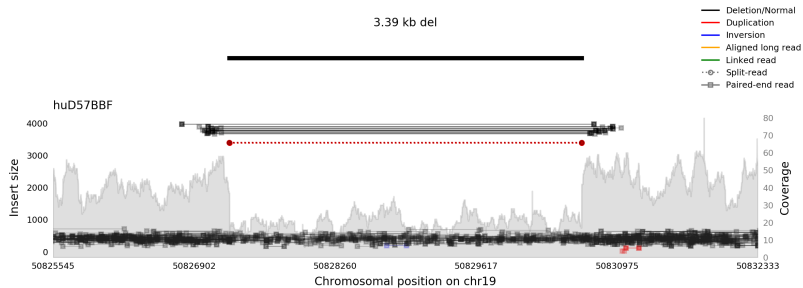
Reference Allele

huD57BBF-sort



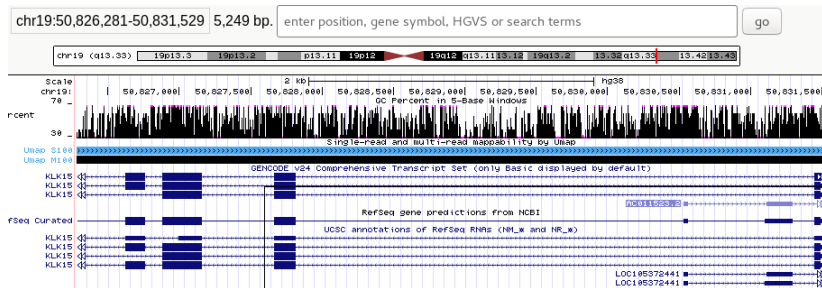
<http://svviz.readthedocs.io>

Viewing deletion – SV-plaudit



<https://github.com/jbelyeu/SV-plaudit>

Genomic region with deletion – KLK15



<http://genome.ucsc.edu/cgi-bin/hgTracks?db=hg38>

KLK15 known function

KLK15

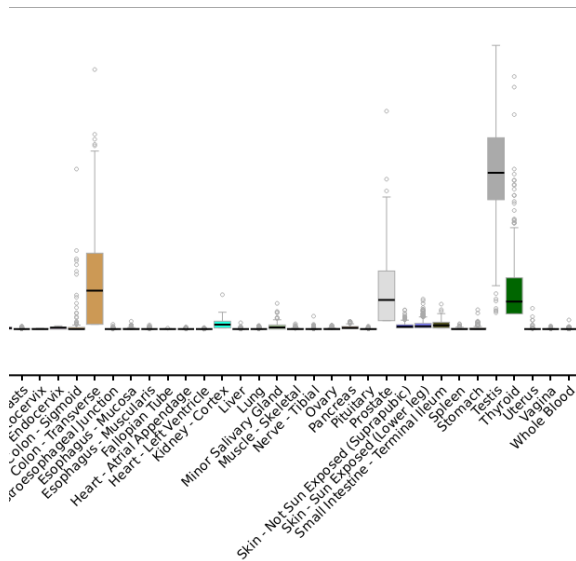
From Wikipedia, the free encyclopedia

Kallikrein-15 is a [protein](#) that in humans is encoded by the *KLK15* [gene](#).^[5]^[6]^[7]^[8]^[9]

Kallikreins are a subgroup of serine proteases having diverse physiological functions. Growing evidence suggests that many kallikreins are implicated in carcinogenesis and some have potential as novel cancer and other disease biomarkers. This gene is one of the fifteen kallikrein subfamily members located in a cluster on chromosome 19. In prostate cancer, this gene has increased expression, which indicates its possible use as a diagnostic or prognostic marker for prostate cancer. The gene contains multiple polyadenylation sites and alternative splicing results in multiple transcript variants encoding distinct isoforms.^[9]

<https://en.wikipedia.org/wiki/KLK15>

Tissue specific gene expression



Self reported conditions

Conditions

Name	Start Date
Benign Prostatic Hypertrophy (BPH)	1998-01-01
Heart murmur	2005-01-01
High Cholesterol	2000-01-01
Thyroid Nodule	2006-01-01

<https://my.pgp-hms.org/profile/huD57BBF>

- Barriers to building analysis pipelines
- bcbio: open source community development
- Common Workflow Language (CWL): assembly language for workflows
- Practical CWL with bcbio: HPC, Cloud, DNAnexus, Arvados, SevenBridges
- Personal Genome Project n=1 analysis example
- **GA4GH: Automating validation and multi-platform testing**

Validation: key component of bcbio

- Pre-built workflows with known outputs
- Cover multiple cases: germline, somatic, low frequency, FFPE, structural variants
- Large collections of diverse workflows

https://github.com/bcbio/bcbio_validation_workflows

- Integration tests for pipelines
- Unbiased algorithm comparisons
- Baseline for improving methods
- Automated tests for platforms



Genome in a Bottle
Consortium



Global Alliance
for Genomics & Health

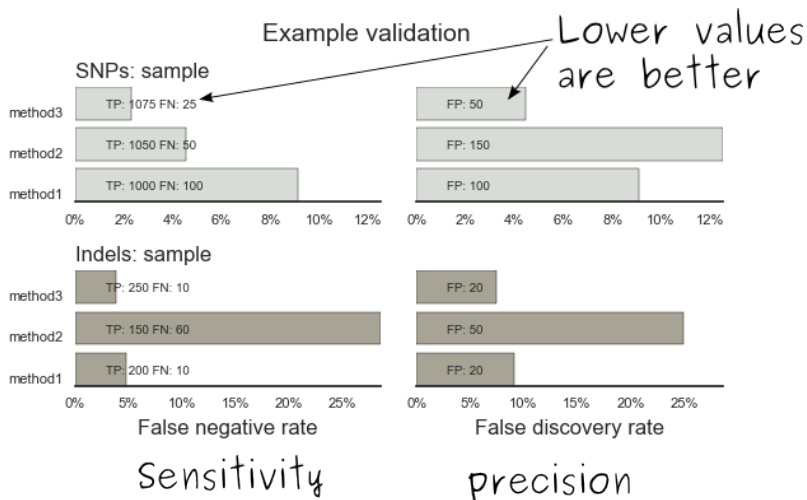
ICGC-TCGA DREAM Mutation Calling challenge

<http://www.genomeinabottle.org/>

<http://ga4gh.org/\#/benchmarking-team>

<https://www.synapse.org/\#!Synapse:syn312572>

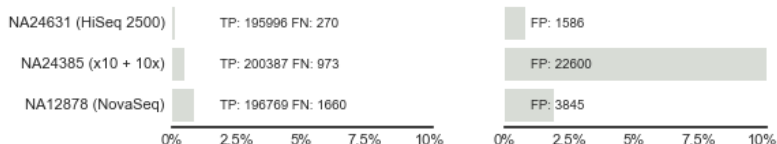
Validation graphs



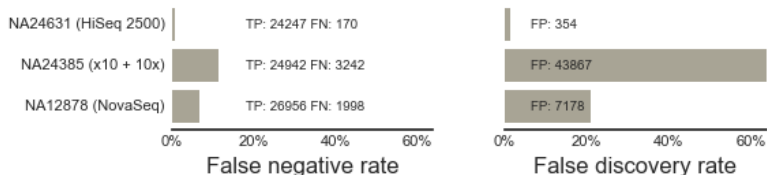
NA12878, NA24385, NA24631 GATK4 joint calling

GATK4 joint calling hg38

SNPs: BQSR + HaplotypeCaller



Indels: BQSR + HaplotypeCaller



https://github.com/bcbio/bcbio_validations/tree/master/gatk4

Need continuous integration process

- Automate testing across multiple platforms
- Test new workflow definitions
- Test new tools and algorithms
- Transparent process

GA4GH: workflow coordination



GA4GH/DREAM Workflow Execution Challenge



Global Alliance
for Genomics & Health



UNIVERSITY OF CALIFORNIA
SANTA CRUZ



Sage
PUBLICATIONS



National Institutes of Health
Turning Discovery Into Health

[https://www.synapse.org/#!Synapse:
syn8507133/wiki/415976](https://www.synapse.org/#!Synapse:syn8507133/wiki/415976)

- Automation of validation
- Workflow Execution Service (WES)
- Shared API for running CWL/WDL workflows
- Contributors welcome

<https://github.com/ga4gh/workflow-execution-schemas>

- Barriers to building analysis pipelines
- bcbio: open source community development
- Common Workflow Language (CWL): assembly language for workflows
- Practical CWL with bcbio: HPC, Cloud, DNAnexus, Arvados, SevenBridges
- Personal Genome Project n=1 analysis example
- GA4GH: Automating validation and multi-platform testing

- Science = collaboration and re-use
- bcbio with interoperable workflow abstractions
- How to run bcbio analyses where you want them
- Interpreting variant calling outputs
- We can build better things together