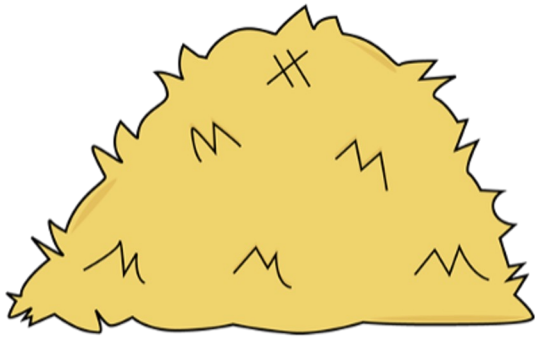
Running workflows on the cloud

Geraldine Van der Auwera, PhD
Broad Institute

September 2022

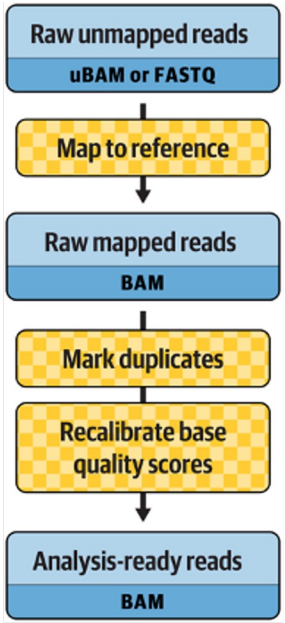
Genomics in a nutshell

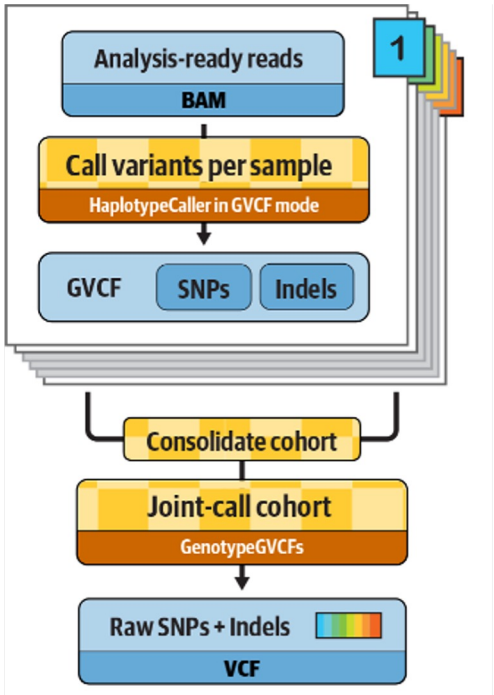


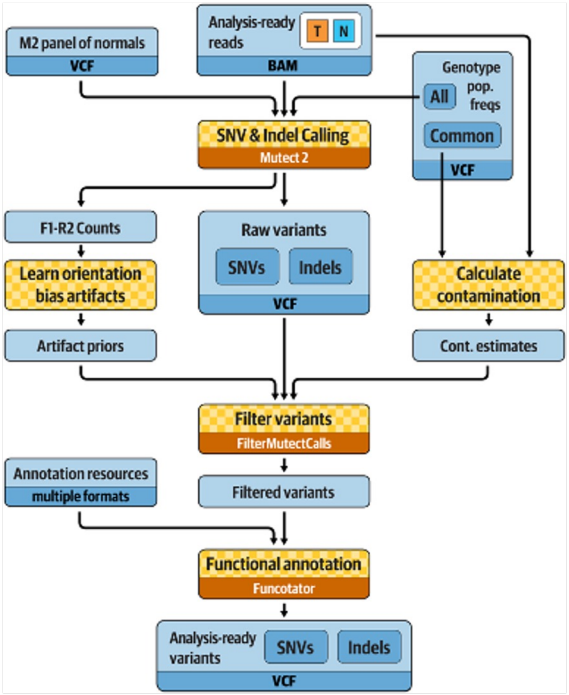
```
Ref  ... T A C A C A T T C A G C ...
Me?  ... T A C G C A T T C A G C ...
```

- 3 Gigabases in a human genome
- High-throughput sequencing
  ⇨ 100 Gb file of short sequences
- 4 to 5 Million small differences
  (relative to standardized reference)
- Which differences matter?

Genomics "pipelines"
= workflows describing series of analysis steps that can be automated







Workflow languages in bioinformatics

Old-school: bash, python

Basically a list of command lines
to run sequentially + control/glue code

New wave of dedicated systems

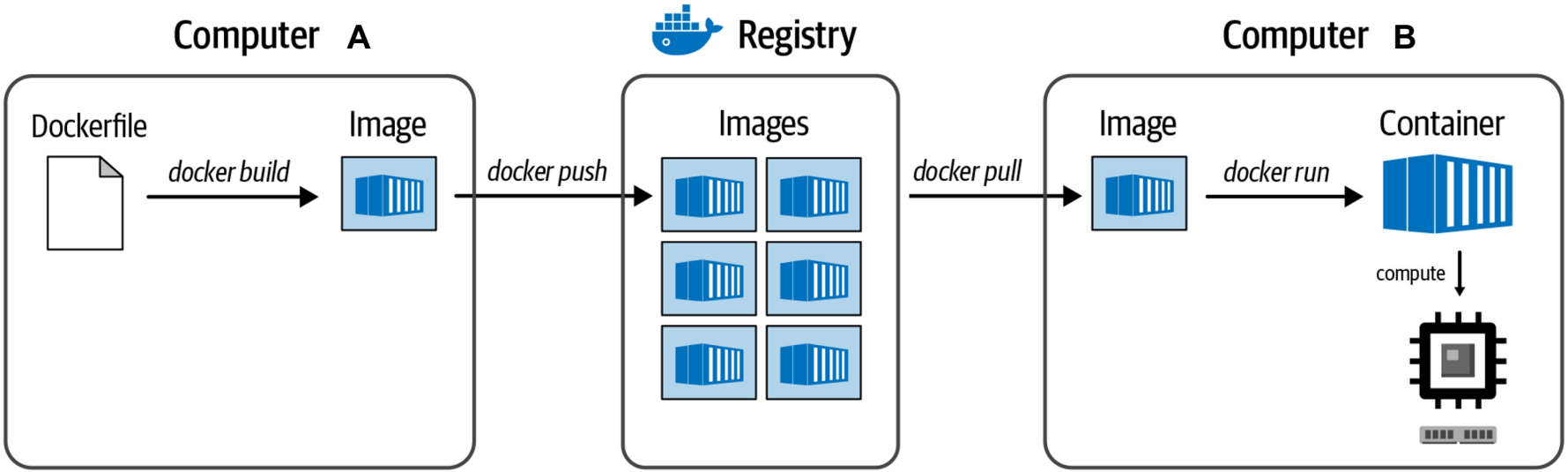
Enforce separation between the analysis work and the logistics of how it gets done

{wdl}

nextflow

COMMON
WORKFLOW
LANGUAGE

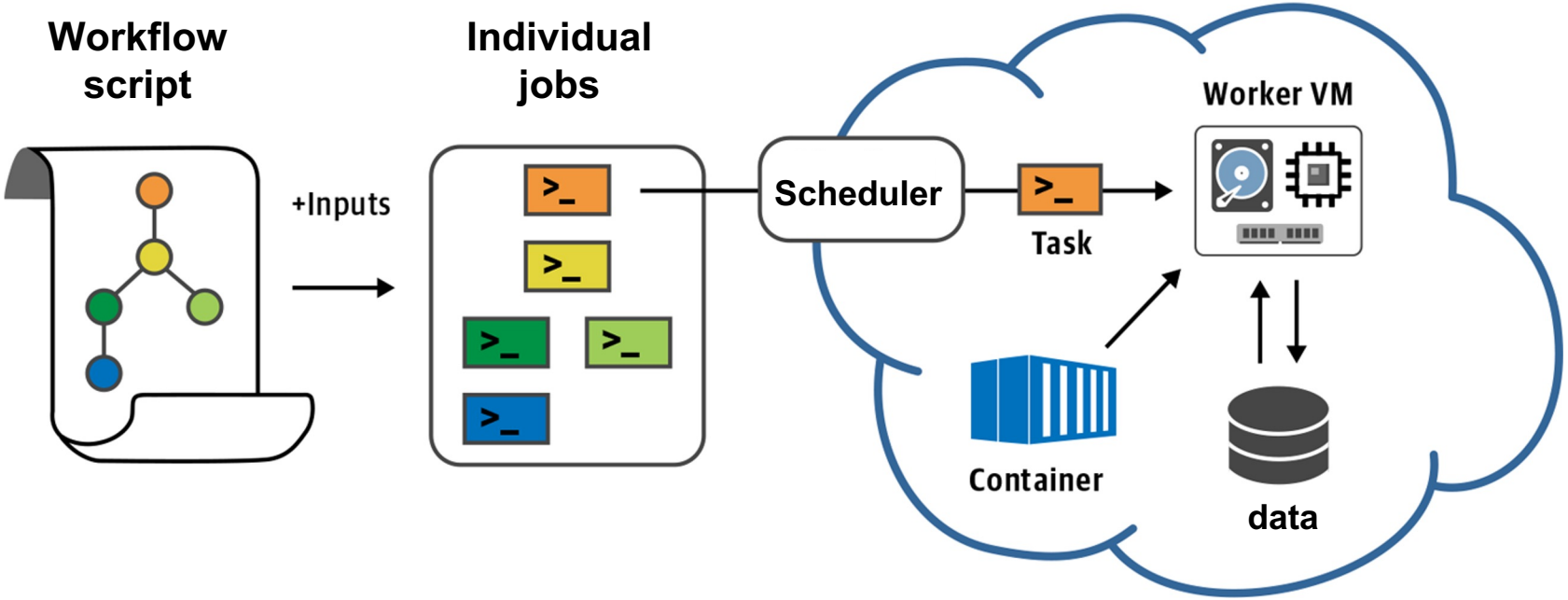
New systems use containers to pull analysis software



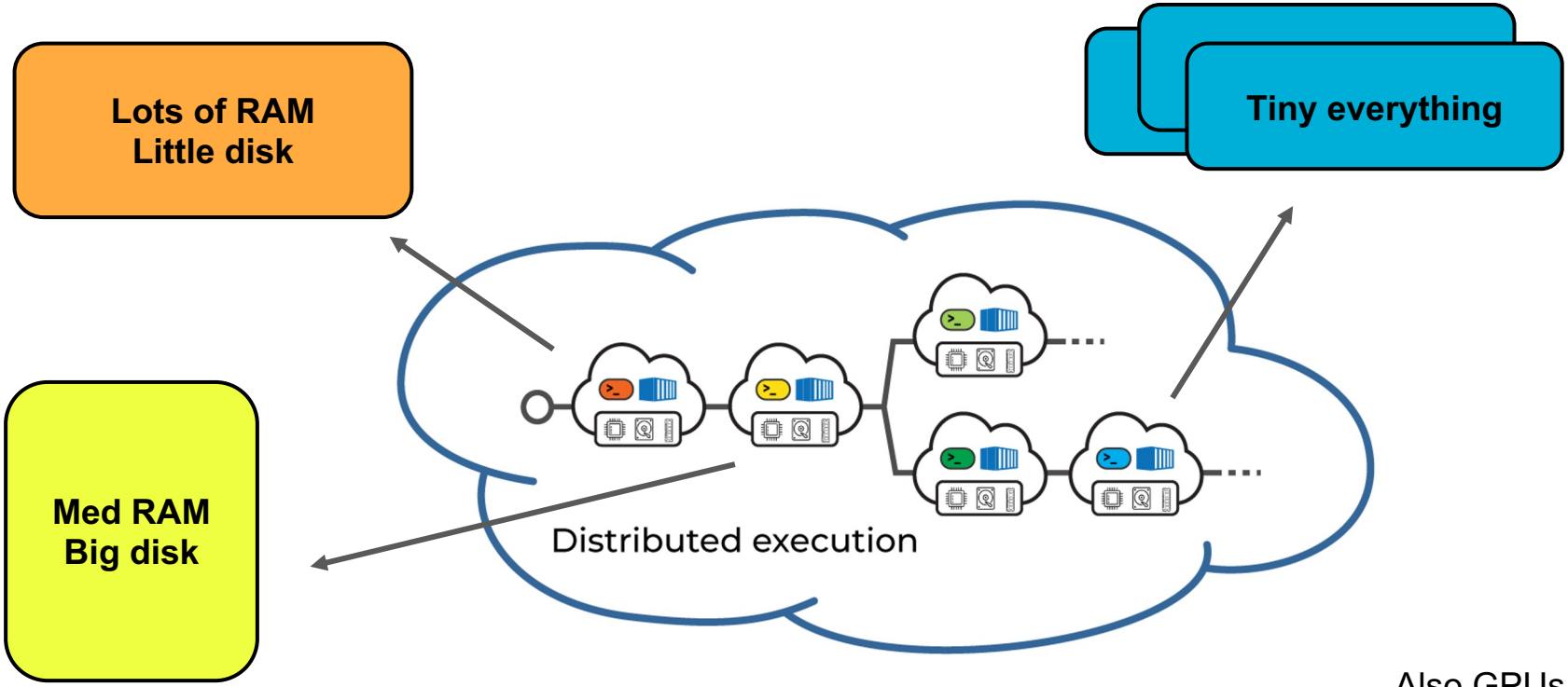
Create an image that encapsulates all necessary software

Use the exact same software environment on a different platform

Easily execute work on separate machines



Dispatch work to appropriately sized machines



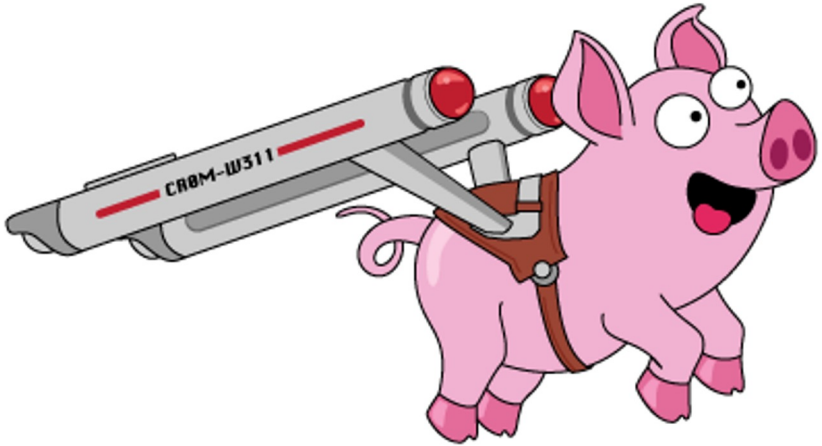
Also GPUs etc

Workflow Description Language

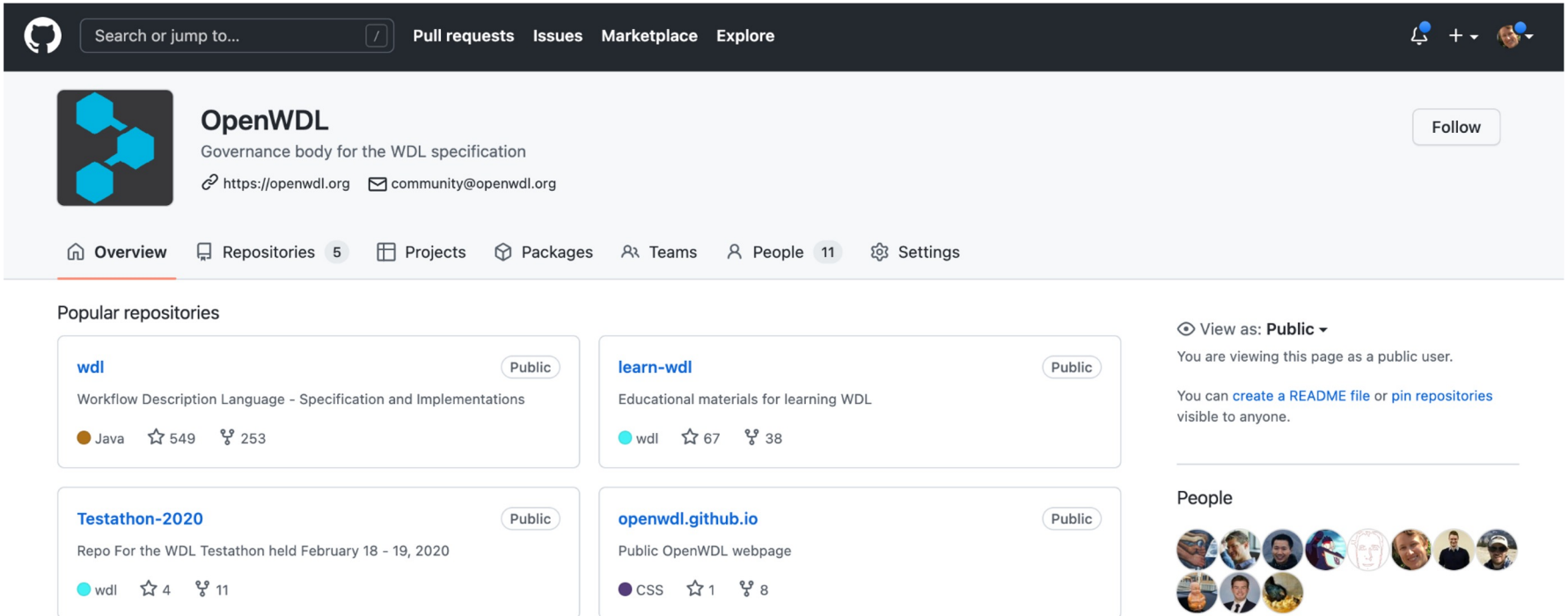
{wdl}

+

Cromwell workflow manager



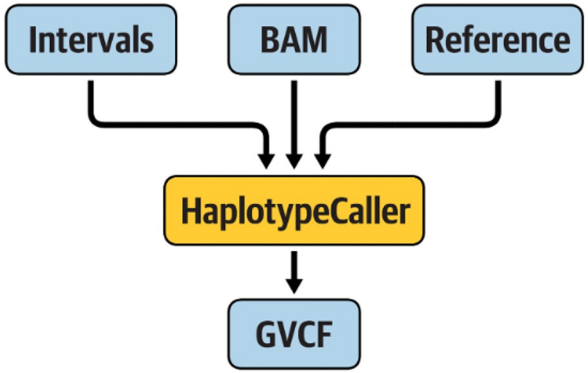
Workflow Description Language is maintained by OpenWDL.org



Basic WDL structure and syntax

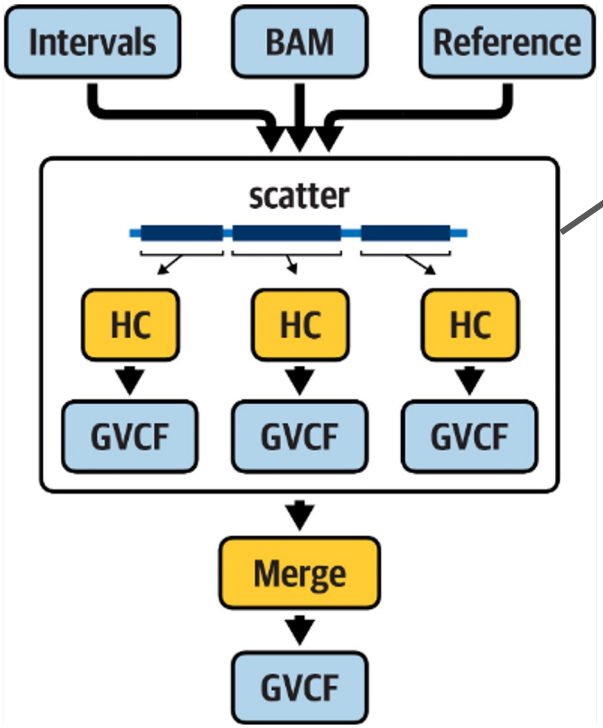
```
task task_A {
  input {
    File ref
    File in
    String id
  }
  command <<<
    do_stuff -R ~{ref} -I ~{in} -O ~{id}.ext
  >>>
  runtime {
    docker: 'my_project/do_stuff:1.2.0'
  }
  output {
    File out = "~{id}.ext"
  }
}
```

```
workflow MyWorkflowName {
  input {
    File my_ref
    File my_input
    String name
  }
  call task_A {
    input:
      ref = my_ref
  }
  call task_B {
    input:
      in = task_A.out
  }
}
task task_A { }
task task_B { }
```




```
workflow HelloHaplotypeCaller {

    call HaplotypeCallerGVCF
}

task HaplotypeCallerGVCF { … }
```

```
command {
        gatk --java-options ${java_opt} HaplotypeCaller \
            -R ${ref_fasta} \
            -I ${input_bam} \
            -O ${gvcf_name} \
            -L ${intervals} \
            -ERC GVCF
    }
```



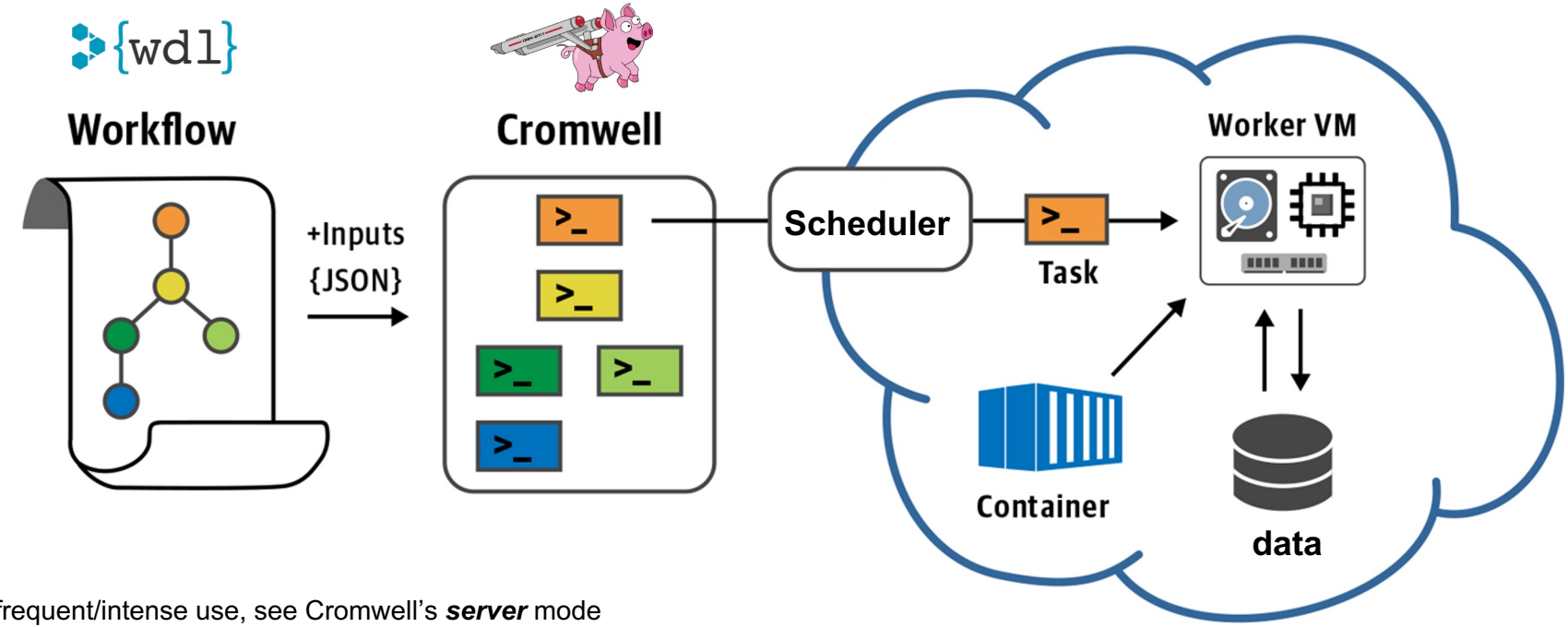

```
scatter(interval in calling_intervals) {
     call HaplotypeCallerGVCF {
          input:
               input_bam = input_bam,
               input_bam_index = input_bam_index,
               intervals = interval,
               gvcf_name = output_basename + ".scatter.g.vcf"
     }
}
call MergeVCFs {
     input:
          vcfs = HaplotypeCallerGVCF.output_gvcf,
          merged_vcf_name = output_basename + ".merged.g.vcf"
}

output {
     File output_gvcf = MergeVCFs.mergedGVCF
}
```

```
task ProcessSomeData {

    inputs {
        Int disk_for_my_task
        … # other inputs
    }

    runtime {
        docker: "broadinstitute/gatk:4.2.0.0"
        memory: "3000 MB"
        disks: "local-disk " + disk_for_my_task + " HDD"
      }

      … # command and output blocks
}
```

Inputs provided through JSON file

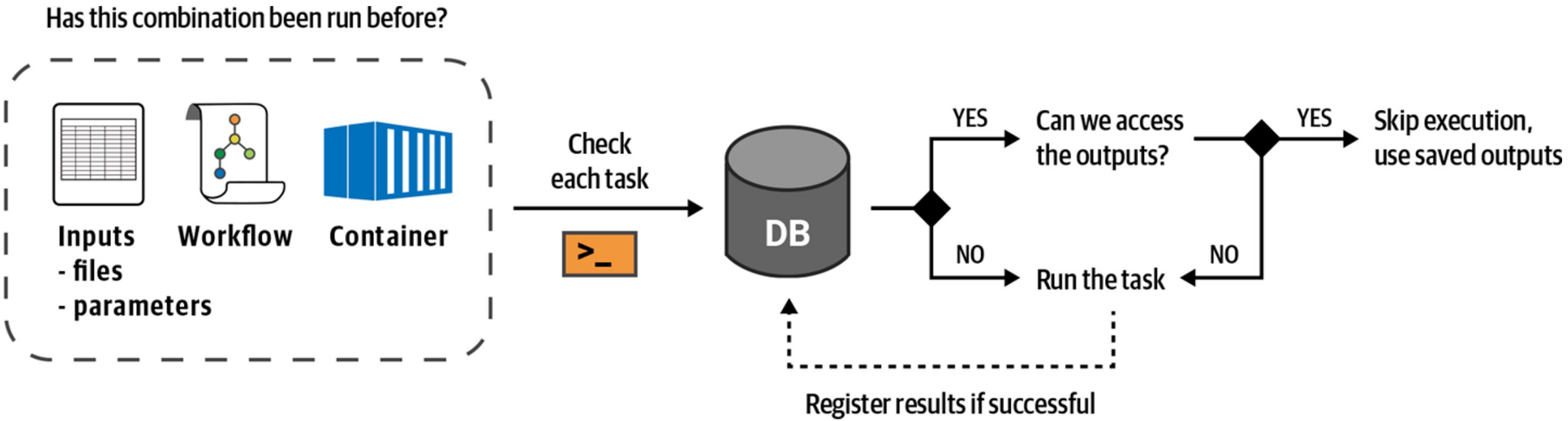
```
{
"HelloHaplotypeCaller.HaplotypeCallerGVCF.input_bam_index": "book/data/germline/bams/mother.bai",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.input_bam": "book/data/germline/bams/mother.bam",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.ref_fasta": "book/data/germline/ref/ref.fasta",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.ref_index": "book/data/germline/ref/ref.fasta.fai",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.ref_dict": "book/data/germline/ref/ref.dict",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.intervals": "book/data/germline/intervals/snippet-intervals-min.list",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.docker_image": "us.gcr.io/broad-gatk/gatk:4.1.3.0",
"HelloHaplotypeCaller.HaplotypeCallerGVCF.java_opt": "-Xmx8G"
}
```

```
java -jar cromwell.jar /
     run haplotypecaller.wdl -i haplotypecaller.inputs.json
```



For frequent/intense use, see Cromwell's ***server*** mode

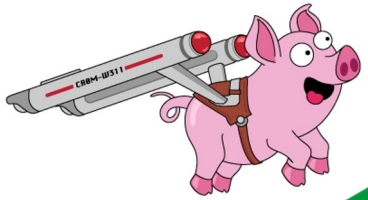
Cromwell has a smart resume feature (call caching)



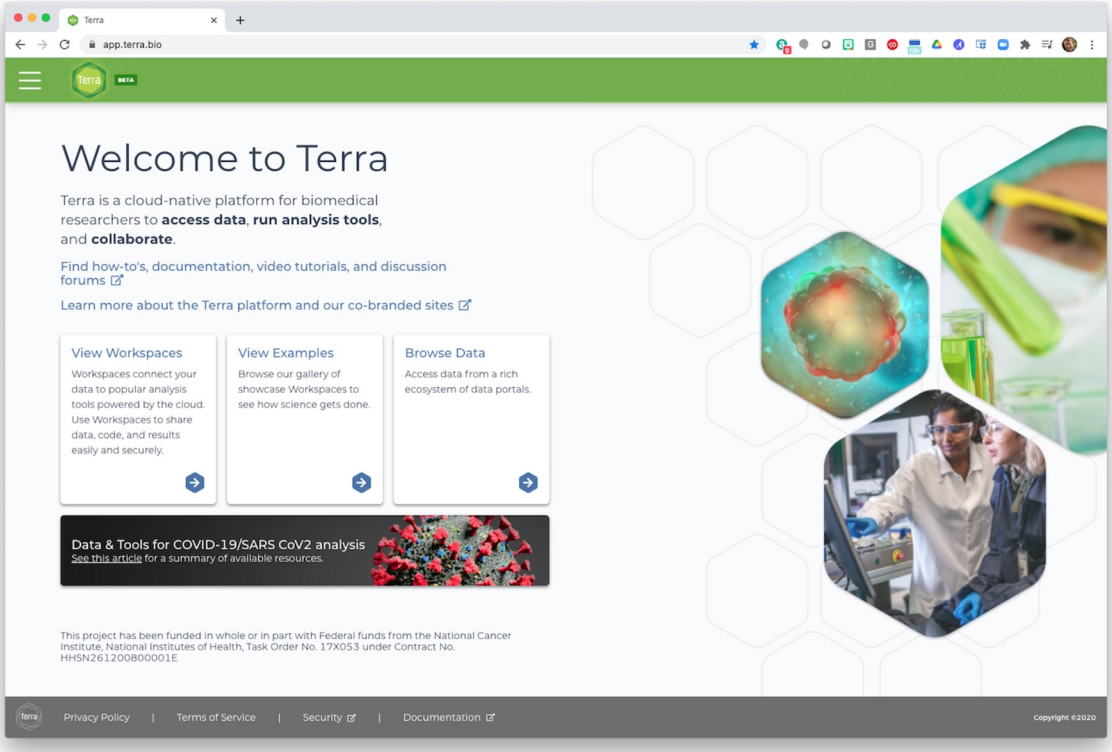
Cromwell backends

- **Local**
- **HPC**, including **Sun Grid Engine**, **LSF**, **HTCondor** & **SLURM**
  - Run jobs as subprocesses or via a dispatcher.
  - Supports launching in Docker containers.
  - Use `bash`, `qsub`, and `bsub` to run scripts.
- **Google Cloud**
  - Launch jobs on Google Compute Engine through the Google Genomics Pipelines API.
- **GA4GH TES**
  - Launch jobs on servers that support the GA4GH Task Execution Schema (TES).
- **AWS Batch (beta)**
  - Use Job Queues on AWS Batch

Cromwell services on the cloud



Terra



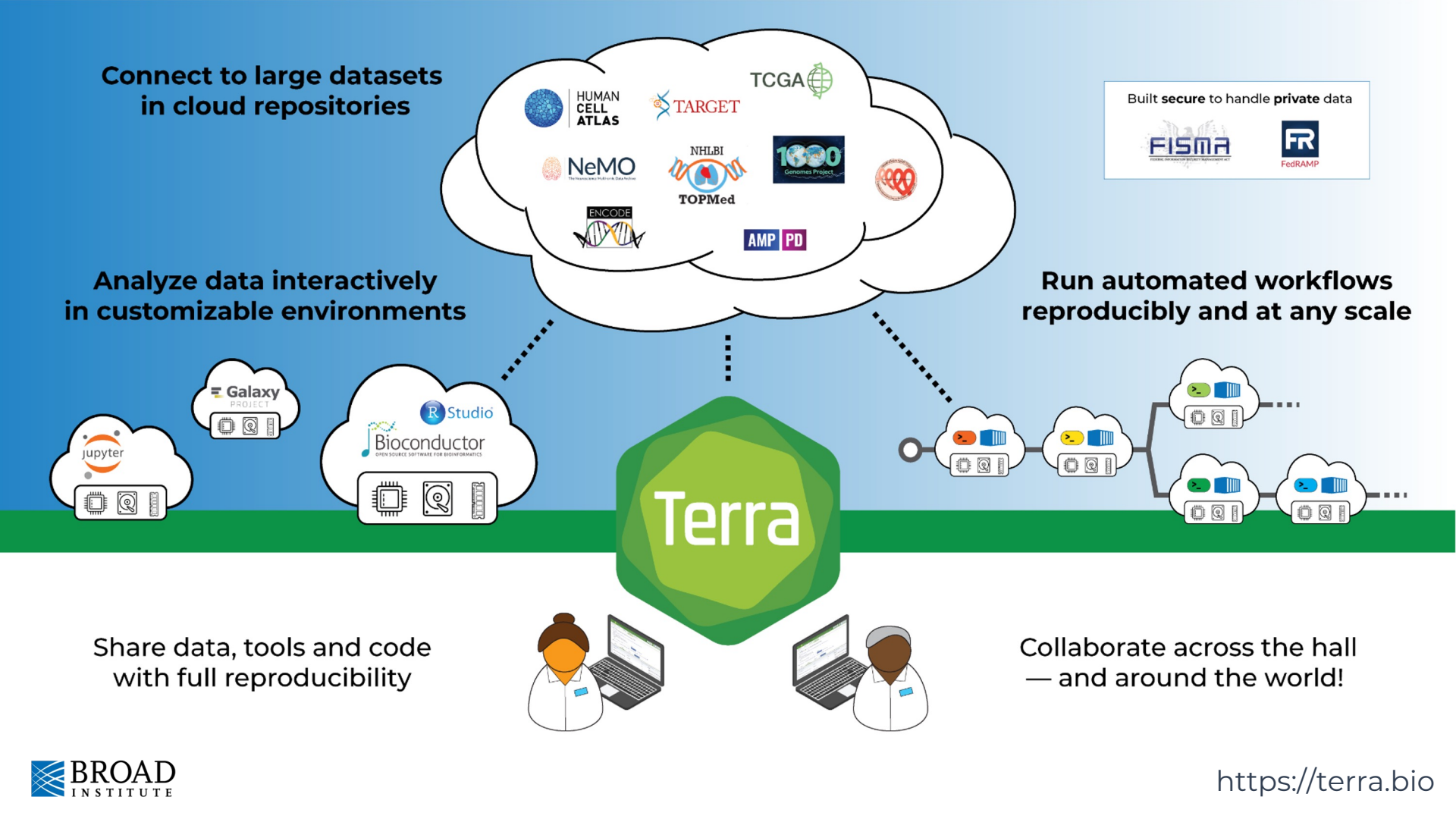
BROAD INSTITUTE

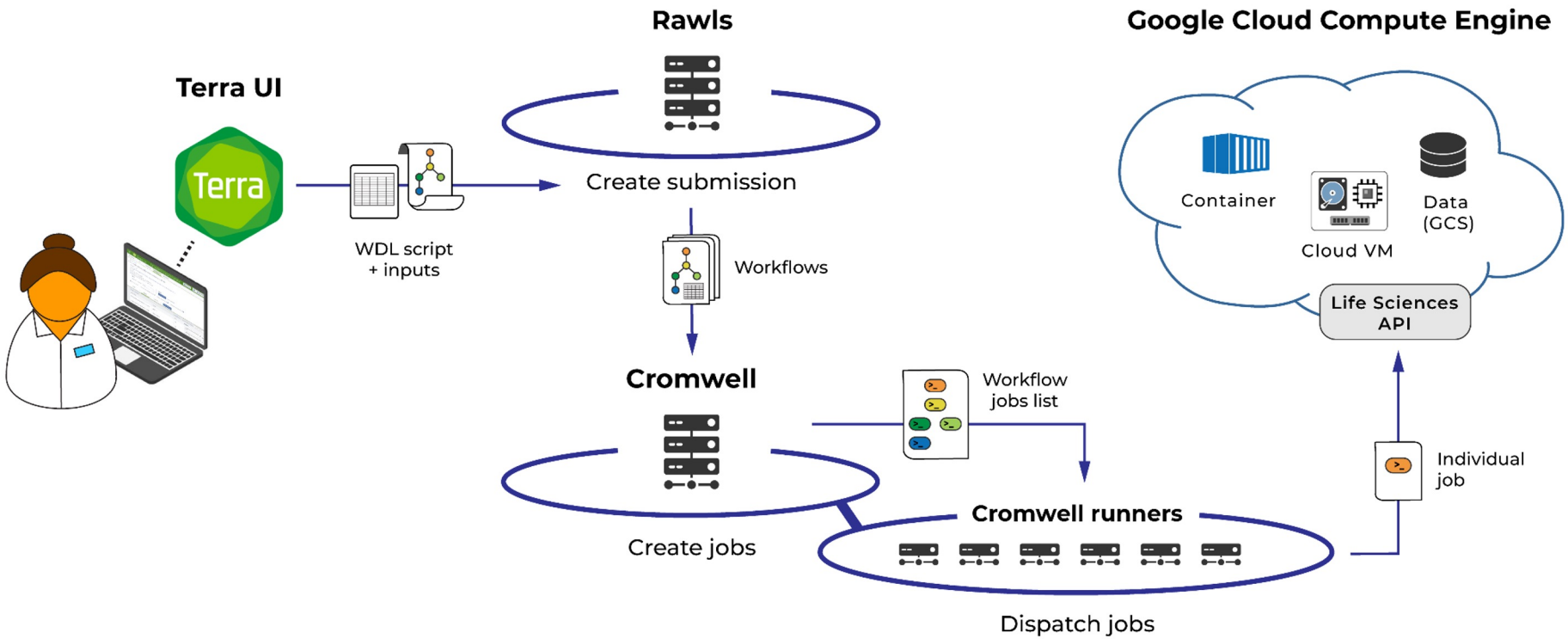
Microsoft

verily

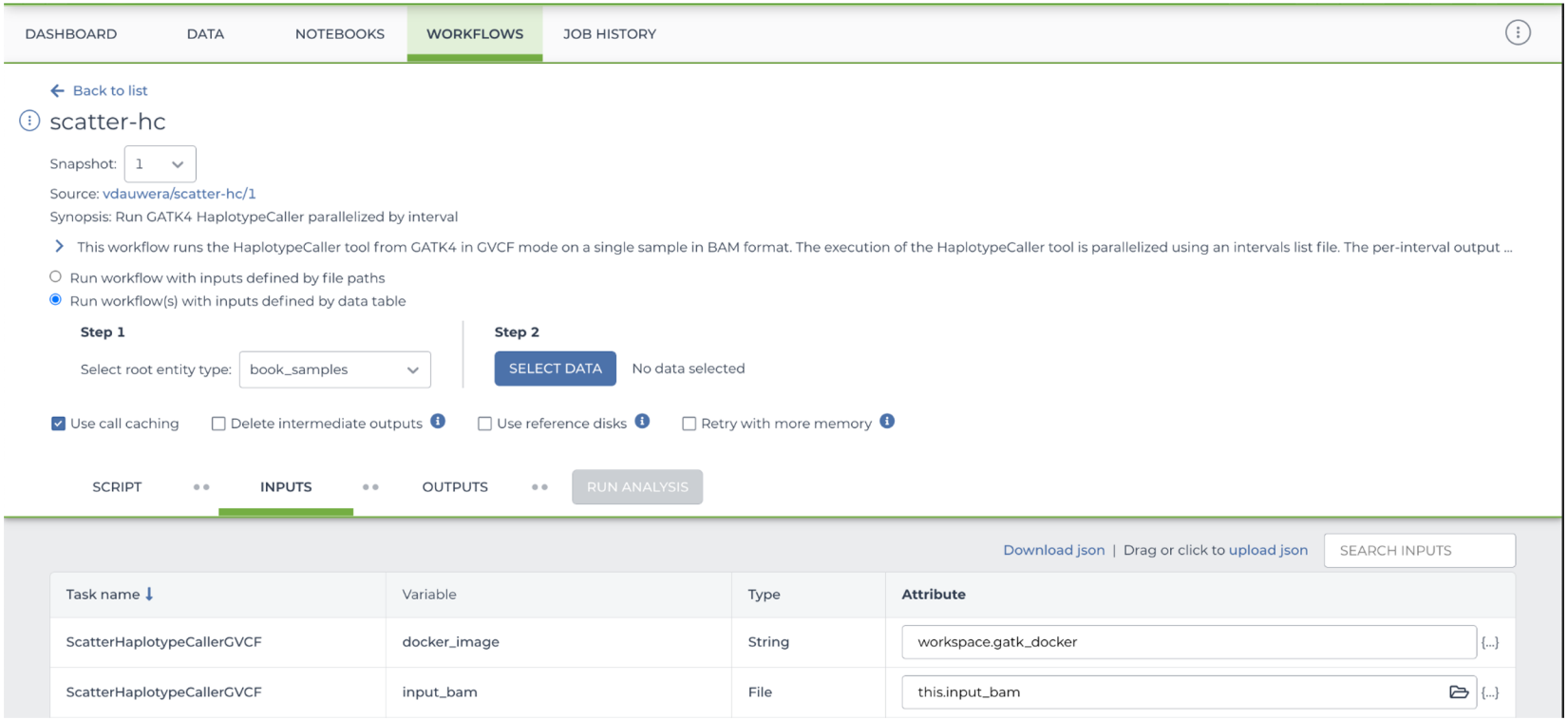
BROAD INSTITUTE

https://terra.bio

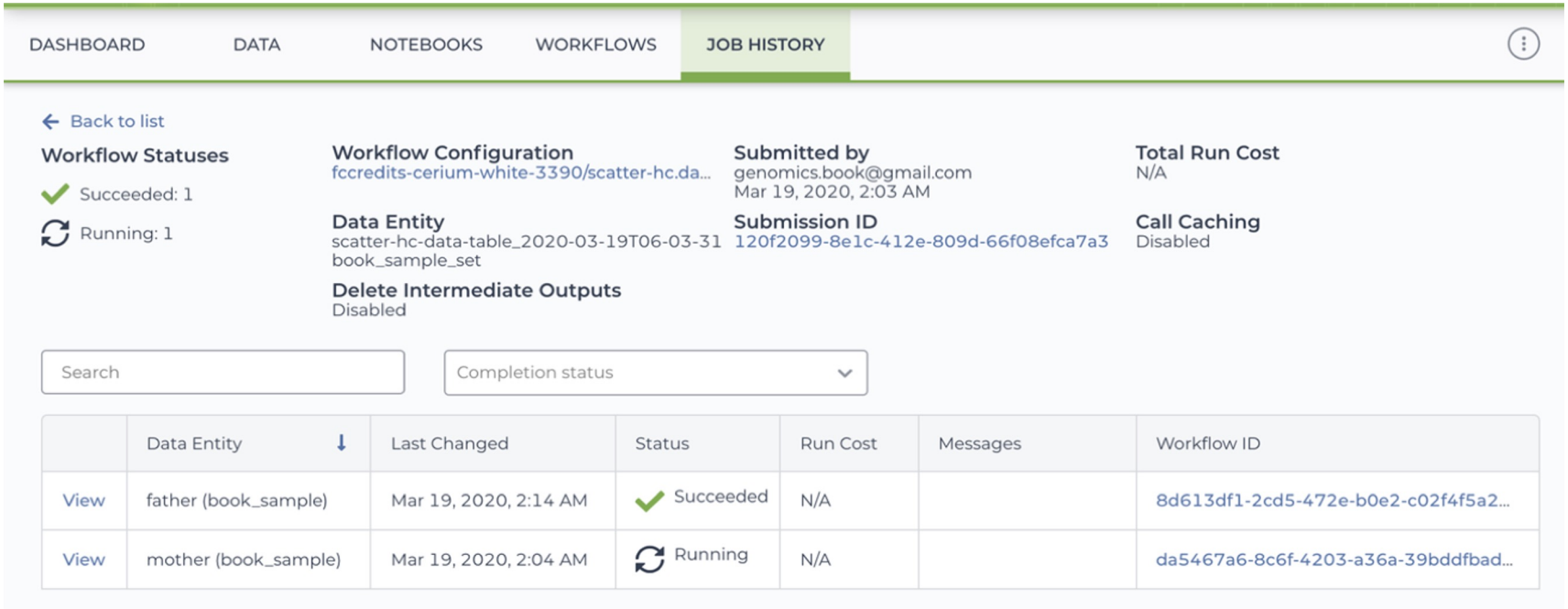

Connect to large datasets in cloud repositories
HUMAN CELL ATLAS
TARGET
TCGA
NeMO
NHLBI
TOPMed
1000 Genomes Project
ENCODE
AMP PD
Built secure to handle private data
FISMA
FedRAMP
Analyze data interactively in customizable environments
jupyter
Galaxy PROJECT
RStudio
Bioconductor
Run automated workflows reproducibly and at any scale
Terra
Share data, tools and code with full reproducibility
Collaborate across the hall — and around the world!
BROAD INSTITUTE
https://terra.bio


Rawls
Google Cloud Compute Engine
Terra UI
Terra
Create submission
WDL script
+ inputs
Workflows
Container
Cloud VM
Data
(GCS)
Life Sciences
API
Cromwell
Workflow
jobs list
Individual
job
Cromwell runners
Create jobs
Dispatch jobs

Graphical interfaces for ease of use



Graphical interfaces for ease of use



Use case #1: T2T variant calling project

A complete reference genome improves analysis of human genetic variation

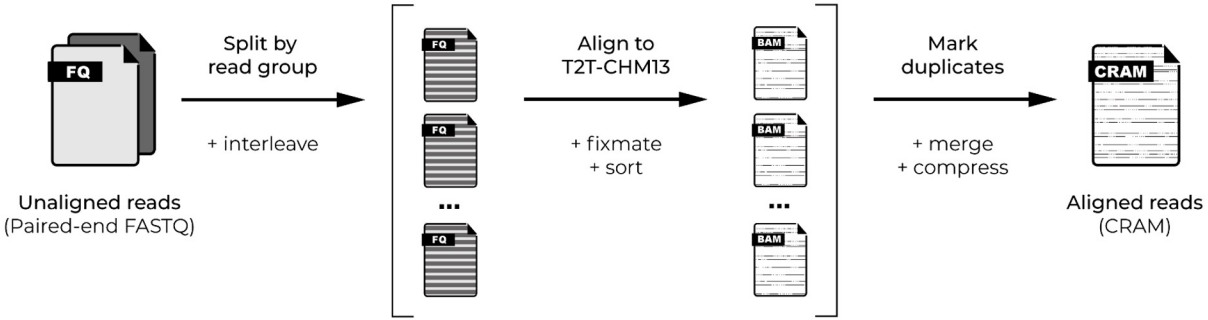
SERGEY AGANEZOV, STEPHANIE M. YAN, DANIELA C. SOTO, MELANIE KIRSCHE, SAMANTHA ZARATE, PAVEL AVDEYEV, DYLAN J. TAYLOR, KISHWAR SHAFIN, ALAINA SHUMATE, [...] MICHAEL C. SCHATZ +24 authors Authors Info & Affiliations

SCIENCE • 1 Apr 2022 • Vol 376, Issue 6588 • DOI: 10.1126/science.abl3533

https://www.science.org/doi/10.1126/science.abl3533

"We show how this reference universally improves read mapping and variant calling for 3202 and 17 globally diverse samples sequenced with short and long reads, respectively."

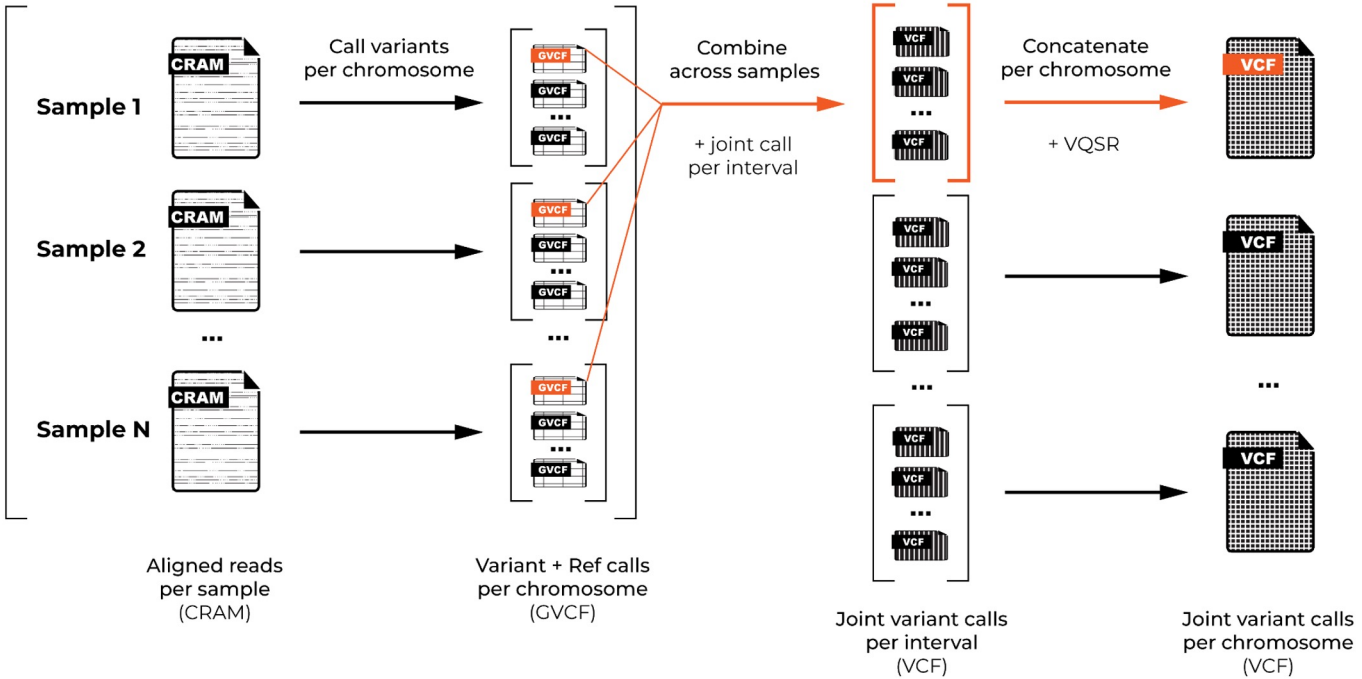
Use case #1: T2T variant calling



Per sample	2 files	~10 files	~10 files	1 file
Total	**6,404 files**	**~32,000 files**	**~32,000 files**	**3,202 files**

Samantha Zarate, Terra Blog
https://terra.bio/calling-variants-from-telomere-to-telomere-with-the-new-t2t-chm13-genome-reference/

Use case #1: T2T variant calling



Per sample	1 file	24 files	NA	NA
Total	**3,202 files**	**>75,000 files**	**~30,000 files**	**24 files**

Use case #1: T2T variant calling / Takeaways

"The push-button capabilities of Terra let us **scale up easily and rapidly**: after **verifying the success of our WDLs on a few samples**, we could move on to **processing hundreds or thousands of workflows at a time**. It took us **about a week** to process everything, and that was with Google's default compute quotas in place (eg max 25,000 cores at a time), which can be raised on request."

Honorable mentions

"We also really appreciated how easy it was to **collaborate with others**"

"More generally, we found that the **reproducibility and reusability** of our analyses have increased significantly."

Samantha Zarate, Terra Blog
https://terra.bio/calling-variants-from-telomere-to-telomere-with-the-new-t2t-chm13-genome-reference/

Use case #2: Cancer proteogenomics

PANOPLY: a cloud-based platform for automated and reproducible proteogenomic data analysis

D. R. Mani ✉, Myranda Maynard, Ramani Kothadia, Karsten Krug, Karen E. Christianson, David Heiman, Karl R. Clauser, Chet Birger, Gad Getz & Steven A. Carr

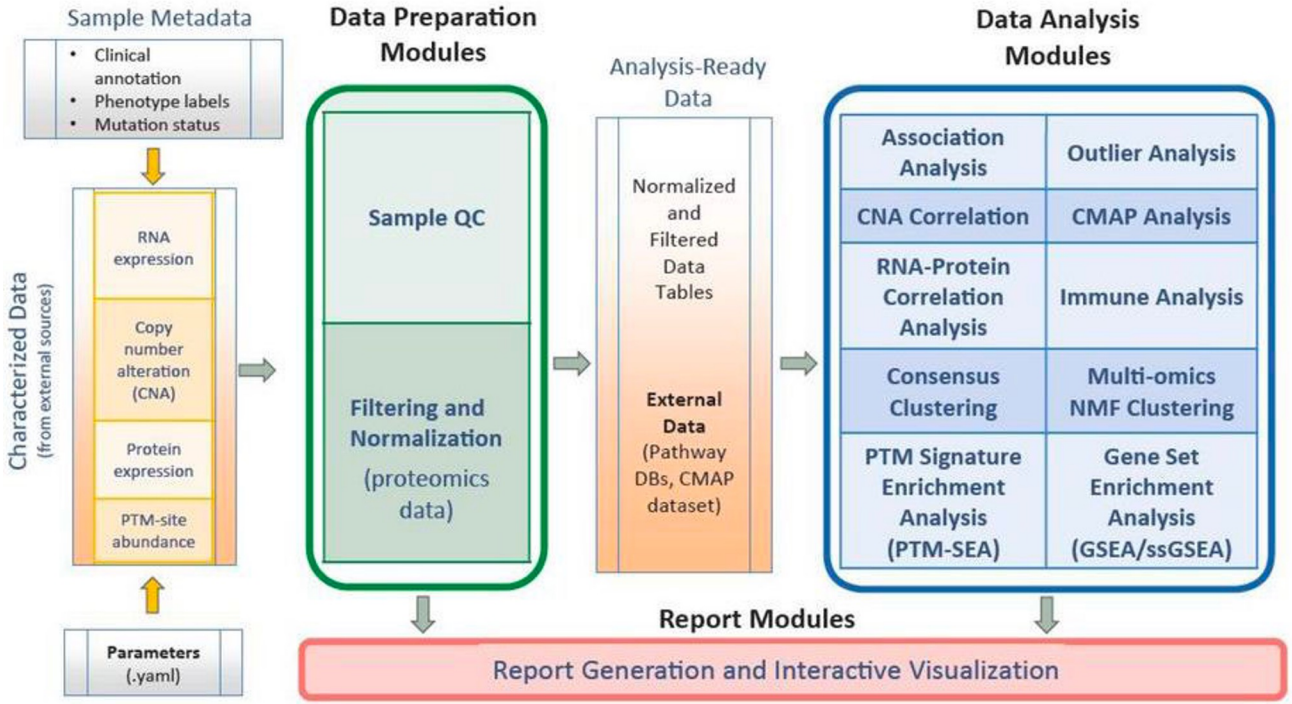
Nature Methods **18**, 580–582 (2021)

https://www.nature.com/articles/s41592-021-01176-6

"PANOPLY uses state-of-the-art statistical and machine learning algorithms to transform multi-omic data from cancer samples into biologically meaningful and interpretable results."

Use case #2: Cancer proteogenomics

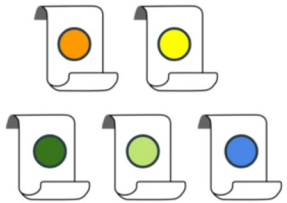
A “greatest hits” compilation of methods from flagship CPTAC studies



Use case #2: Cancer proteogenomics / Takeaways

"We recognized that **enabling a wide range of people** to use PANOPLY, especially those with **less computational experience**, would require **more than just releasing code**. We wanted a way to make PANOPLY **usable out of the box** [...]."

DR Mani, Terra Blog https://terra.bio/panoply-framework-for-cancer-proteogenomics/



Modules workspace

Separate workflow per module

Maximum flexibility

Can compose new pipelines
Can add new modules

Pipelines workspace

Unified pipelines for standard use cases

Maximum reproducibility

Can run exactly as published

Tutorial workspace

Pre-run clone of the Pipelines WS

Includes preconfigured dataset (BRCA)

Job history shows execution results reproducing parts of Mertins *et al*, 2016*

Use case #3: Pathogen genomic surveillance

Phylogenetic analysis of SARS-CoV-2 in Boston highlights the impact of superspreading events

JACOB E. LEMIEUX, KATHERINE J. SIDDLE, BENNETT M. SHAW, CHRISTINE LORETH, STEPHEN F. SCHAFFNER, ADRIANNE GLADDEN-YOUNG, GORDON ADAMS, TIMELIA FINK, CHRISTOPHER H. TOMKINS-TINCH, [...] BRONWYN L. MACINNIS +44 authors Authors Info & Affiliations

SCIENCE • 10 Dec 2020 • Vol 371, Issue 6529 • DOI: 10.1126/science.abe3261

https://www.science.org/doi/10.1126/science.abe3261

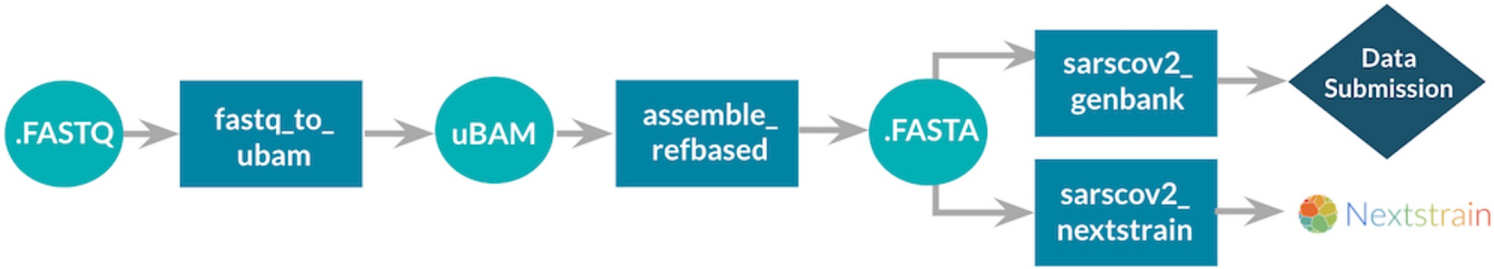
Transmission from vaccinated individuals in a large SARS-CoV-2 Delta variant outbreak

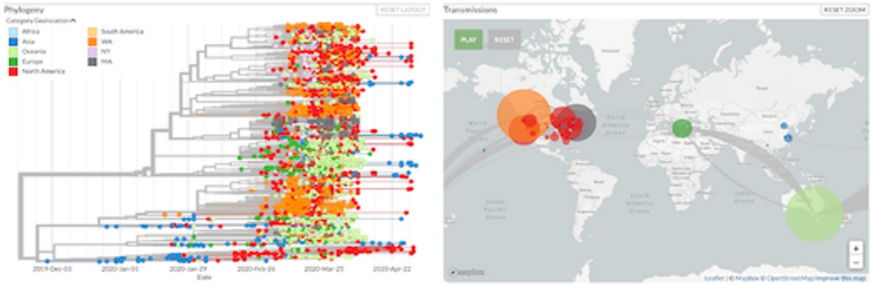
Katherine J. Siddle [16, 18] • Lydia A. Krasilnikova [16] • Gage K. Moreno [16] • ... Daniel J. Park [17] • Bronwyn L. MacInnis [17] • Pardis C. Sabeti [17] • Show all authors • Show footnotes

Open Access • Published: December 22, 2021 • DOI: https://doi.org/10.1016/j.cell.2021.12.027

https://doi.org/10.1016/j.cell.2021.12.027

Use case #3: Pathogen genomic surveillance





Terra COVID-19 workspace
https://app.terra.bio/#workspaces/pathogen-genomic-surveillance/COVID-19

Use case #3: Pathogen genomic surveillance / takeaways

"Once we understood the processes, [we] began **porting their existing workflows** from DNAnexus and GitHub to Terra."

"Unsurprisingly, in working with such large amounts of data, there were some initial hiccups and challenges; for example, we had to figure out **how to organize the data effectively**, and we ran into some **Google Cloud usage quotas**"

"We hope this will **empower public health labs**, as they scale their viral sequencing work."

Christine Loreth, Terra Blog
https://terra.bio/behind-the-scenes-bringing-the-analysis-of-covid-19-data-from-greater-boston-into-the-cloud/

See also:
https://terra.bio/new-partnership-with-cdc-boosts-terra-support-for-public-health-labs-across-the-usa/

WDLs in the mist

github.com/broadinstitute/gatk/cnv_somatic_pair_workflow:4.2.0.0

☆ 1

Last Modified: 5 days ago

Info | Launch | Versions | Files | Tools

Workflow Information

Source Code: github.com/broadinstitute/gatk:4.2.0.0

TRS: #workflow/github.com/broadinstitute/gatk/cnv_somatic_pair_workflow

Topic: Official code repository for GATK versions 4 and up

Checker Workflow: n/a

Descriptor Type: WDL

DOI: n/a

Workflow Version Information

4.2.0.0 Export as ZIP

Launch with

Warning: this version of the WDL has imports, which are not supported by DNAstack. Make sure to select a version without imports in DNAstack.

DNAnexus »

Terra »

eLwazi »

AnVIL »

NHLBI BioData Catalyst

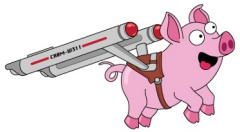
Recent Versions

4.2.0.0 Feb 19, 2021

master Aug 27, 2022

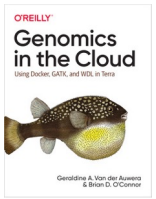
Resources

OpenWDL repositories on Github
https://github.com/openwdl



Cromwell docs on Github
https://cromwell.readthedocs.io/

Terra.bio WDL Resources
https://support.terra.bio/hc/en-us/sections/360007274612-WDLs-Resources



Genomics in the Cloud book
https://oreil.ly/genomics-cloud