

Intro to Dockstore

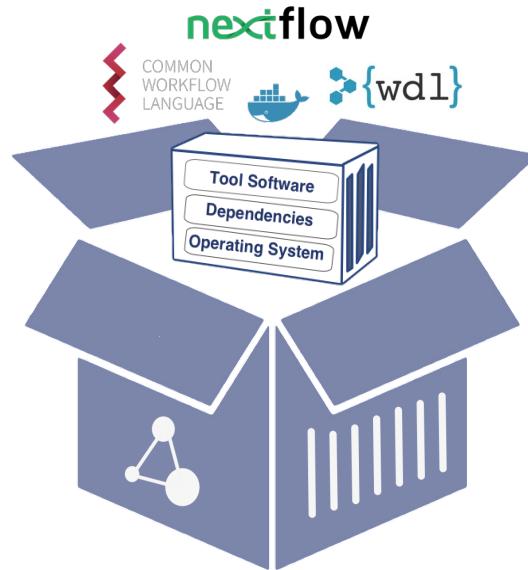
Louise Cabansay

UC Santa Cruz, Genomics Institute
Software Engineer, Dockstore

What is Dockstore?

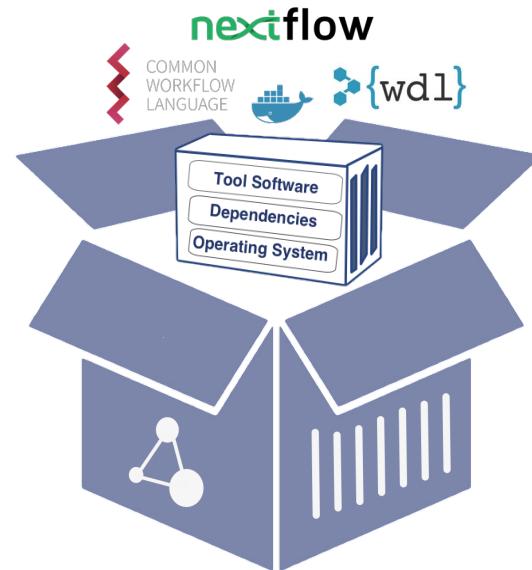
Dockstore is a free and open source platform for sharing scientific tools and workflows. It is a registry of Docker-based resources described using popular workflow languages CWL, WDL, and Nextflow.

- **Portability**
 - Run workflows in any environment that supports Docker
- **Interoperability**
 - Standardize computational analysis through GA4GH APIs
- **Reproducibility**
 - Create, Share, Use
 - Containers + Popular descriptor languages



What is Dockstore? (simple, tl;dr)

- A platform for sharing research software
- An ‘app store’ for bioinformatics



How is Dockstore useful? What problems does it solve?

Simplest Use Case: Search and Findability

A researcher wants to do some type of genomics work (assembly, alignment, variant calling, GWAS, etc.)

- What software can be used for the job?
- What works with their data?
- Where do they download the software? What version?

Dockstore as a Registry

Searchable and Centralized catalogue of bioinformatics tools and workflows

- Many ways to get tools and workflows into Dockstore!
 - External Hosting: Github, Bitbucket, Gitlab, Quay.io, Docker Hub
 - Direct Hosting (descriptors)



Search Functionality

- Search by name, author, organization, descriptor, etc
- Filter results using facets
 - Descriptor Language
 - Labels

Advanced Search

Find tools and workflows with files ▾ that have ...

All these words: e.g. bcbio

The exact word or phrase: e.g. cgp

Any of these words: e.g. wxs

None of these words: e.g. wgs

Clear All Advanced Search

Share Search: contains one of "Topmed"

Browse Tools Browse Workflows

Tag Cloud

A workflow is a series of tools strung together, with an associated descriptor describing how to run it.

Name	Author	Format	Project Links	Stars
DataBiosphere/topmed-workflows/UM_variant_caller_wd	Walt Shands	WDL	GitHub	6★
DataBiosphere/topmed-workflows/UM_aligner.wdl	Walt Shands	WDL	GitHub	2★
DataBiosphere/topmed-workflows/CCDG_aligner_functional	Ruchi Munshi	WDL	GitHub	2★
DataBiosphere/topmed-workflows/UM_aligner_cwl	Seven Bridges	CWL	GitHub	1★
DataBiosphere/topmed-workflows/gatk-vcf-comparator	n/a	CWL	GitHub	
DataBiosphere/topmed-workflows/CCDG_aligner_functional	Yilin Xu	CWL	GitHub	
DataBiosphere/topmed-workflows/UM_variant_caller_cwl	Vladimir Obucina	CWL	GitHub	

Items per page: 10 | < > | 1 - 7 of 7

Organizations and Collections

- Organizations
 - A place for groups, labs, consortiums, etc to showcase their projects
 - Markdown descriptions
 - Membership roles
- Collections
 - Group of tools/workflows into highlighted by an organization
 - Markdown descriptions

The screenshot shows the AnVIL Analysis, Visualization and Informatics Lab-space interface. At the top, there is a navigation bar with icons for search, organizations, and docs, and a login button. Below the navigation bar, the word "Organizations" is displayed. The main content area features the AnVIL logo and title. A sub-header states: "The AnVIL is a project funded by the NIH to create a managed platform for genomics researchers". Below this, a link to the website (<https://anvilproject.org/>) and a star rating section are shown. The interface includes tabs for "Collections", "Members", and "Events". Under the "Collections" tab, two items are listed: "GATK4" (Variant caller) and "Single Cell Pipelines" (Collection of Single Cell processing pipelines). To the right of the collections list, there is a detailed sidebar with sections for "About the Project", "Project Aims", "Operate services", and "Organize and host key NHGRI datasets".

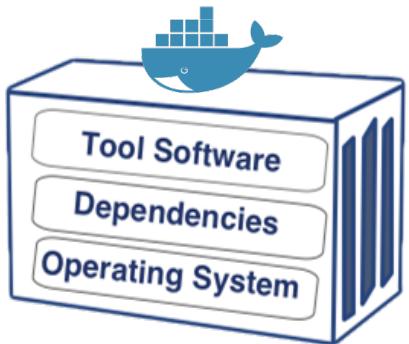
Common Use Case: Simplifying Installation Process

A researcher already knows what software they want to use, but...

- Installation problems
 - Software was built on a different OS (executable files don't run)
 - Other problems: compiler or build issues
- Dependency problems
 - Software requires different version than one installed on machine (ex. Java, Python)
 - Multiple tools need the same dependency, but different versions of that dependency

How It Works: Packaging Content

Container



+

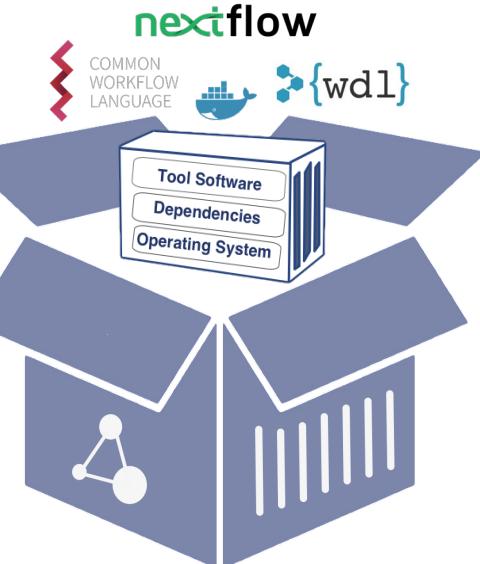
Descriptor



COMMON
WORKFLOW
LANGUAGE



nextflow



Software is “packaged” using containerization technology and described using descriptor language

- Analysis can be moved from cloud-to-cloud, VM-to-VM, server-to-server and yet be guaranteed to run on anything that supports Docker

Containers, Descriptors, and Parameter Files

Containers, Descriptors, and Parameter Files

Container:

Packaged up code with its all dependencies. This allows for portable software that runs quickly and reliably from one computing environment to another.



Containers, Descriptors, and Parameter Files

Container:

Packaged up code with its all dependencies. This allows for portable software that runs quickly and reliably from one computing environment to another.



Descriptor:

The workflow language used to *describe* how to run your tool or workflow/pipeline.

- Which tools/containers
- What steps and when
- Define parameters
 - I/O data
 - compute requirements
- Metadata



Containers, Descriptors, and Parameter Files

Container:

Packaged up code with its all dependencies. This allows for portable software that runs quickly and reliably from one computing environment to another.



Descriptor:

The workflow language used to *describe* how to run your tool or workflow/pipeline.

- Which tools/containers
- What steps and when
- Define parameters
 - I/O data
 - compute requirements
- Metadata



Parameter File (wdl, cwl):

- Specifies the actual input/output files (local, ftp, http, or cloud)
- Set compute resources
- JSON, YAML

```
{  
    "bam_input": {  
        "class": "File",  
        "format": "http://edamontology.org/format_2572",  
        "path": "/tmp/NA12878.chrom20.ILLUMINA.bwa.CEU.low_coverage.20121211.bam"  
    },  
    "bamstats_report": {  
        "class": "File",  
        "path": "/tmp/bamstats_report.zip"  
    }  
}
```

Container Simple Example

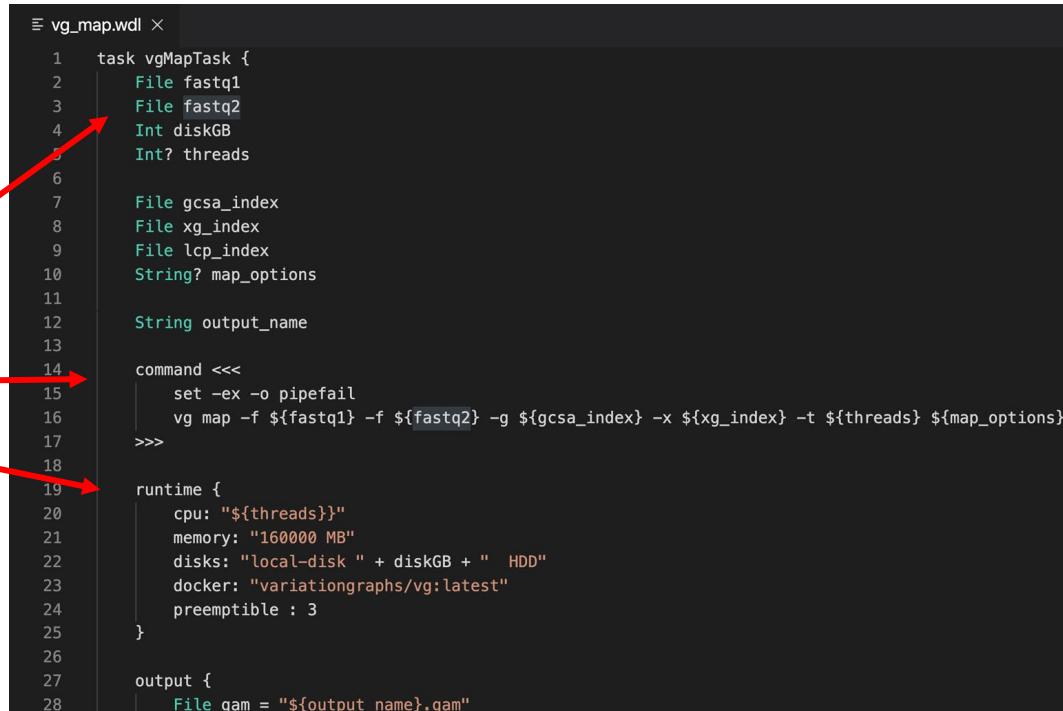
- Images of containers are built from dockerfiles which describe the packaged up environment:
 - Operating System or Base Image to build upon
 - Dependencies to install
 - Scripts or commands to run to complete environment set-up

```
FROM ubuntu:18.04
MAINTAINER jmonlong@ucsc.edu
RUN apt-get update \
&& apt-get install -y --no-install-recommends \
wget \
bcftools \
tabix \
gcc \
time \
python \
python-pip \
python-setuptools \
locales \
make \
python-dev \
&& rm -rf /var/lib/apt/lists/*
RUN pip install --upgrade pip
RUN pip install awscli
WORKDIR /home
RUN wget --no-check-certificate https://github.com/dellytools/delly/ \
&& mv delly_v0.8.1_parallel_linux_x86_64bit /bin/delly \
&& chmod +x /bin/delly
```

Descriptor Simple Example (wdl task snippet)

Descriptor: A workflow language used to *describe* how to run your tool or workflow/pipeline.

- Define workflow into tasks and what order to run those tasks (aka a script)
- Define parameters
 - Input and output files
 - Parameterized commands
 - Compute requirements
- Metadata
 - Authorship, contact information, etc



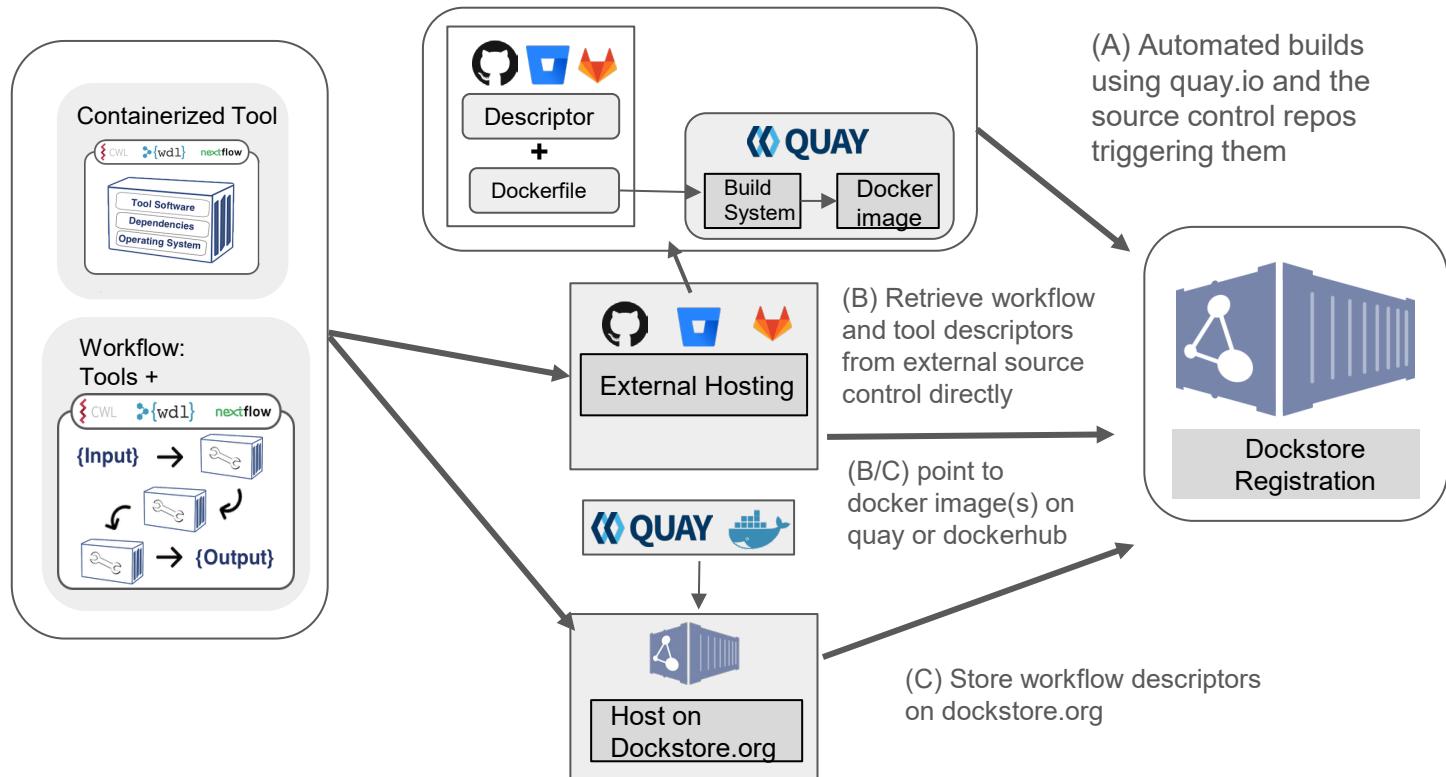
```
vg_map.wdl x
1 task vgMapTask {
2   File fastq1
3   File fastq2
4   Int diskGB
5   Int? threads
6
7   File gcsa_index
8   File xg_index
9   File lcp_index
10  String? map_options
11
12  String output_name
13
14  command <<<
15    set -ex -o pipefail
16    vg map -f ${fastq1} -f ${fastq2} -g ${gcsa_index} -x ${xg_index} -t ${threads} ${map_options}
17  >>>
18
19  runtime {
20    cpu: "${threads}"
21    memory: "160000 MB"
22    disks: "local-disk " + diskGB + " HDD"
23    docker: "variationgraphs/vg:latest"
24    preemptible : 3
25  }
26
27  output {
28    File gam = "${output_name}.gam"
```

Parameter File Simple Example:

- Specify the actual input and actual files/values to use when running tasks
- File provisioning plugins allow for data to be fetched from cloud storage (s3, gs, etc.)

```
1   "vg_map.INPUT_READ_FILE_1": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/reads_1.fastq.gz",
2   "vg_map.INPUT_READ_FILE_2": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/reads_2.fastq.gz",
3   "vg_map.SAMPLE_NAME": "test_sample_SV",
4   "vg_map.XG_FILE": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/vg-sv-ex.xg",
5   "vg_map.GCSA_FILE": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/vg-sv-ex.gcsa",
6   "vg_map.GCSA_LCP_FILE": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/vg-sv-ex.gcsa.lcp",
7   "vg_map.SNARL_FILE": "https://jmonlong.s3-us-west-2.amazonaws.com/public/vg-sv/vg-sv-ex.snarls",
8   "vg_map.VG_CONTAINER": "quay.io/vgteam/vg:v1.19.0",
9   "vg_map.READS_PER_CHUNK": 10000000,
10  "vg_map.SPLIT_READ_CORES": 1,
11  "vg_map.SPLIT_READ_DISK": 10,
12  "vg_map.MAP_CORES": 1,
13  "vg_map.MAP_DISK": 10,
14  "vg_map.MAP_MEM": 8,
15  "vg_map.MERGE_GAM_CORES": 1,
16  "vg_map.MERGE_GAM_DISK": 10,
17  "vg_map.MERGE_GAM_MEM": 8,
18  "vg_map.MERGE_GAM_TIME": 2400,
19  "vg_map.VGCALL_CORES": 1,
20  "vg_map.VGCALL_DISK": 10,
21  "vg_map.VGCALL_MEM": 8,
22  "vg_map.GOOGLE_CLEANUP_MODE": "true",
23  "vg_map.CLEANUP_MODE": "false"
24
25 }
```

Ways to Register to Dockstore

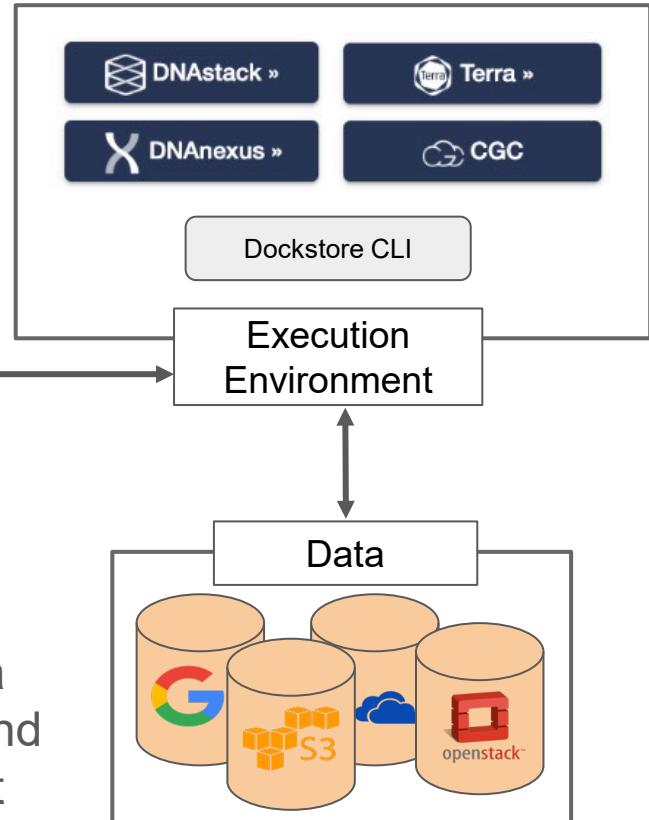
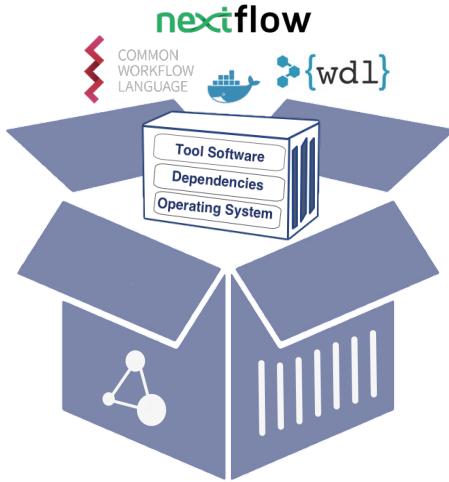


Complex Use Cases: Collaborations and Scale

Ex: A team of collaborators from 8 different institutions are working together on a study that analyzes the genomes of thousands of individuals. But...

- Each institution uses a different compute environment for analysis
 - Workstations, local machines, or small servers for development
 - Large analysis on various cloud platforms or different custom HPCs
- Pipeline/Script is designed to work for one institution's data but can't easily be interchanged or doesn't scale properly at another
- Everything seems to have installed, but the output differs between compute environments

Solution: Interoperable Cloud Ecosystem



Standardized APIs allow
Dockstore to connect to a
variety cloud execution and
data storage environment



Available Workflows

github.com/DataBiosphere/topmed-workflows/UM_aligner_wdl:1.32.0

Last Modified: 19 days ago

datacommons datastage topmed u-of-michigan

Info Launch Versions Files Tools

Descriptor Files Test Parameter Files

u_of_michigan_aligner.wdl

```

256 echo "Running pre-alignment"
257
258 samtools view -T ${ref_fasta} -uh -F 0x900 ${input_cram} \
| bam-ext-mem-sort-manager squeeze --in .ubam --keepDups --rmTags AS:i,
| samtools sort -l 1 -@ 1 -n -T ${pre_output_base}.samtools_sort_tmp - \
| samtools fixmate - - \
| bam-ext-mem-sort-manager bam2fastq --in .bam --outBase ${pre_output_b
263
264 }
265 output {
266   File output_list_file = "${pre_output_base}.list"
267   # Capture all the files mentioned in the pre_output_base.list file
268   # So they will be present for the Align task
269   Array[File] output_fastq_gz_files = glob("${pre_output_base}.*")
270 }
271 runtime {
272   maxRetries: max_retries
273   preemptible: preemptible_tries
274   #memory: "6.5 GB"
275   memory: sub(memory, "\\\\"*) + " GB"
276   cpu: sub(CPUs, "\\\\"*)
277   disks: "local-disk" + sub(disk_size, "\\\\"*) + " HDD"
278   zones: "us-central1-a us-central1-b us-east1-d us-central1-c us-central1-f
279   docker: docker_image
280 }
281
283 task Align {
284   File input_list_file
285   Array[File] input_fastq_gz_files
287
288   File ref_alt
289   File ref_bwt
290   File ref_pac
291   File ref_maf
  
```

Recent Versions

- 1.32.0 Mar 11, 2019
- ✓ 1.29.0 Sep 27, 2018
- feature/freeze8-wdl Jun 24, 2019
- master Mar 11, 2019
- develop Mar 11, 2019
- feature/variant-discovery-task Nov 11, 2018
- See all versions

Source Repositories

GitHub

Collections

TOPMed Alignment Pipeline

Launch with

DNAstack

DNAnexus

Terra

CGC : Cancer Genomics Cloud (Seven Bridges)

BioData Catalyst

Launching Analysis

- Partner Platforms

- DNAstack
- DNAnexus
- Terra
- CGC : Cancer Genomics Cloud (Seven Bridges)
- BioData Catalyst

Launch with





BETA

IMPORT WORKFLOW

Importing from Dockstore

github.com/vgteam/vg_wdl/vg_map_call_sv
V.spack

 Please note: Dockstore cannot guarantee that the WDL and Docker image referenced by this Workflow will not change. We advise you to review the WDL before future runs.

```
1 version 1.0
2
3 workflow vgMapCallSV {
4     meta {
5         author: "Jean Monlong"
6             email: "jmonlong@ucsc.edu"
7             description: "Read mapping and SV genotyping using vg. It takes
8     }
9     input {
10         String SAMPLE_NAME
11             # The sample name
12         File? INPUT_READ_FILE_1
13             # Input sample 1st read
14         File? INPUT_READ_FILE_2
15             # Input sample 2nd read
16         File? INPUT_CRAM_FILE
17             # Input CRAM file
18         File? CRAM_REF
19             # Genome fasta file asso
20         File? CRAM_REF_INDEX
21             # Index of the fasta fil
22         String VG_CONTAINER = "quay.io/vgteam/vg:v1.19.0" # VG Contain
23         File XG_FILE
24             # Path to .xg index file
25         File GCSA_FILE
26             # Path to .gcsa index fi
27         File GCSA_LCP_FILE
28             # Path to .gcsa.lcp inde
29         File? SNARL_FILE
30             # (OPTIONAL) Path to sna
31         File? GBWT_FILE
32             # (OPTIONAL) Path to .gb
33         File? PATH_LIST_FILE
34             # (OPTIONAL) Text file w
```

Destination Workspace

Dockstore-Webinar

1000G_Workspace_Test

AnVIL_CCDG_WashU_CVD_EOCAD_BioImage_WG

Copy of GWAS Notebook using HAIL

Dockstore-Webinar

Test-Workspace



UNIVERSITY OF CALIFORNIA
SANTA CRUZ | Genomics
Institute





DASHBOARD

DATA

NOTEBOOKS

WORKFLOWS

JOB HISTORY

[← Back to list](#)

vg_map_call_sv

Version: svpack

Source: github.com/vgteam/vg_wdl/vg_map_call_sv:svpack

Synopsis:

No documentation provided

 Process single workflow from files Process multiple workflows from: Select data type... Select Data Use call caching

SCRIPT

INPUTS

OUTPUTS

RUN ANALYSIS

Show optional inputs

[Download json](#) | [Drag or click to upload json](#)[SEARCH INPUTS](#)

Task name	Variable	Type	Attribute	
vgMapCallSV	GCSA_FILE	File	"gs://jmonlong-vg-wdl-dev-test/vg-sv-ex.gcsa"	
vgMapCallSV	GCSA_LCP_FILE	File	"gs://jmonlong-vg-wdl-dev-test/vg-sv-ex.gcsa.lcp"	
vgMapCallSV	SAMPLE_NAME	String	"test_sample_SV"	
vgMapCallSV	XG_FILE	File	"gs://jmonlong-vg-wdl-dev-test/vg-sv-ex.xg"	

vgMapCallsV ID: 1f250e26-0737-43ba-80ca-f4c6ffa4c34a 

workspace-id: ef6c094d-3fe6-4a74-bc67-4cf7e2afb2c1 submission-id: a1838031-ca4e-45c7-beee-580315bed431

Status: Succeeded 

Tasks: 6 succeeded, 0 failed, 0 currently being processed

Submitted: Today, 3:46 PM

Started: Today, 3:47 PM

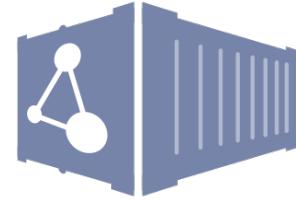
Ended: Today, 3:58 PM (0h 11m)

LIST VIEW	INPUTS	OUTPUTS	LABELS	TIMING DIAGRAM	Status	Start	Duration	Inputs	Outputs	Links	Attempts
Task Name						Today, 3:47 PM	0h 2m	 	  	1	
extractPathNames						Today, 3:47 PM	0h 3m	 	  	1	
firstReadPair						Today, 3:47 PM	0h 2m	 	  	1	
secondReadPair						Today, 3:47 PM	0h 2m	 	  	1	
runVGMAP						Today, 3:50 PM	0h 2m			 	1
mergeAlignmentGAMChu...						Today, 3:53 PM	0h 2m	 	  	1	
runVGPackCaller						Today, 3:55 PM	0h 2m	 	  	1	

Dockstore CLI

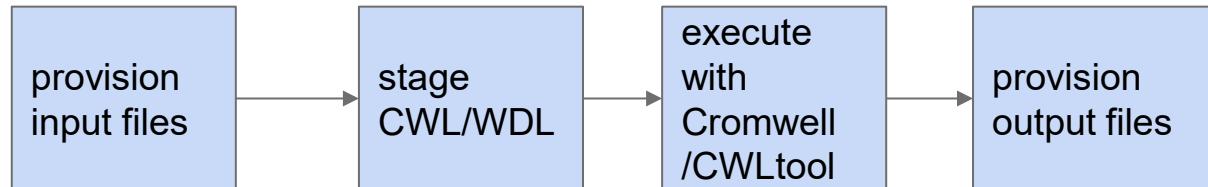
A handy command line resource for users to run content locally

- Pulls tool/workflow from dockstore
- File provisioning
- Execute locally with Cromwell or CWLtool



```
dockstore workflow launch --entry  
github.com/vgteam/vg_wdl/vg_map_call_sv:svpack --json parameter.json
```

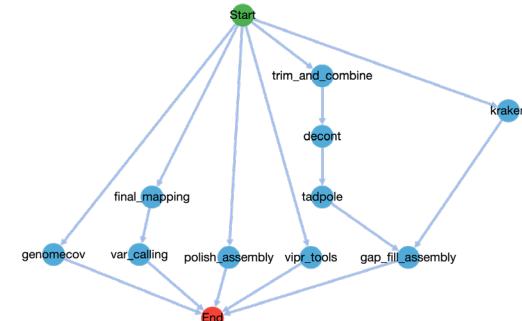
Execution with the Dockstore Command Line Interface (CLI)



Simple Dockstore Command Line

Other Dockstore Features

- Language Support
 - Syntax Highlighting and Validation
 - DAG visualization
- Snapshot and DOI generation
 - Freeze workflow versions using our new snapshot feature
 - Request DOIs for frozen workflows via Zenodo to use as publication references
- Prototype for Apps and Services now public!



Brief Developer Details



Source Code and API

<https://dockstore.org/>

[dockstore / dockstore](https://github.com/dockstore/dockstore)

Watch 27 ⚡ Unstar 64 Fork 18

Code Issues 460 Pull requests 4 Projects 0 Wiki Security Insights

Our VM/Docker sharing infrastructure and management component <https://dockstore.org/>

dockstore docker cwl wdl containers bioinformatics nextflow workflow

2,158 commits 24 branches 178 releases 1 environment 22 contributors Apache-2.0

Branch: develop New pull request Create new file Upload files Find file Clone or download

NatalieEO Feature/2498/track tos acceptance (#2626) ... Latest commit 8916d3e 5 hours ago

Author	Commit Message	Time Ago
.circleci	Feature/2374/token resource it (#2581)	15 days ago
.github	Feature/dockstore classpath (#342)	3 years ago
THIRD-PARTY-NOTICES	Standardize model boilerplate (#1356)	last year
dockstore-client	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-common	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-event-consumer	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-file-plugin-parent	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-integration-testing	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-language-plugin-parent	[maven-release-plugin] prepare for next development iteration	2 days ago
dockstore-webservice	Feature/2498/track tos acceptance (#2626)	5 hours ago
openapi-java-client	[maven-release-plugin] prepare for next development iteration	2 days ago
openapi-java-wes-client	[maven-release-plugin] prepare for next development iteration	2 days ago
reports	[maven-release-plugin] prepare for next development iteration	2 days ago

Github: <https://github.com/dockstore/dockstore>

Swagger <https://dockstore.org/swagger.json> Explore

Dockstore API 1.8.1

Base (https://dockstore.org/api/v1/)

This describes the dockstore API, a webservice that manages pairs of Docker images and associated metadata such as CWL documents and Dockerfiles used to build those images

Terms of service
Dockstore@ga4gh - Website
Send email to Dockstore@ga4gh
Apache License Version 2.0
Dockstore documentation

Schemes: HTTPS Authorize

Filter by tag

entries Interact with entries in Dockstore regardless of whether they are containers or workflows

- POST /entries/{id}/aliases Add aliases linked to a entry in Dockstore.
- GET /entries/{id}/collections Get the collections and organizations that contain the published entry
- POST /entries/{id}/topic Create a discourse topic for an entry.

containers List and register entries in the dockstore (pairs of images + metadata (CWL and Dockerfile))

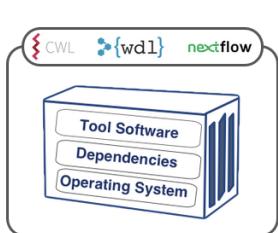
- GET /containers/{alias}/aliases Retrieves a tool by alias.
- GET /containers/{containerId} Retrieve a tool.
- PUT /containers/{containerId} Update the tool with the given tool.
- DELETE /containers/{containerId} Delete a tool.
- GET /containers/{containerId}/descriptor/{relative-path} Get the corresponding descriptor file.

Swagger: <https://dockstore.org/api/static/swagger-ui/index.html#>

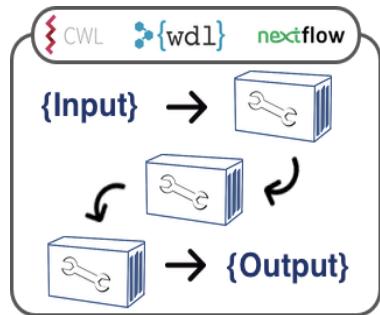
GA4GH Tool Registry Service (TRS) API

The Global Alliance for Genomics and Health (GA4GH) Tool Registry API standard for listing and describing available tools for exchange, indexing, and searching.

- Dockstore implements TRS for launching tools/workflows into external analysis platforms



or



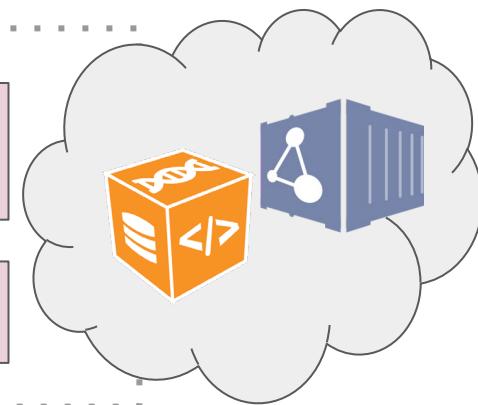
Workflow: Tools + Descriptor

Containerized Tool
+ Descriptor

Sharing API

GET /api/ga4gh/v2/tools/
/{id}/versions/{ver
sion_id}

GET /api/ga4gh/v2/tools

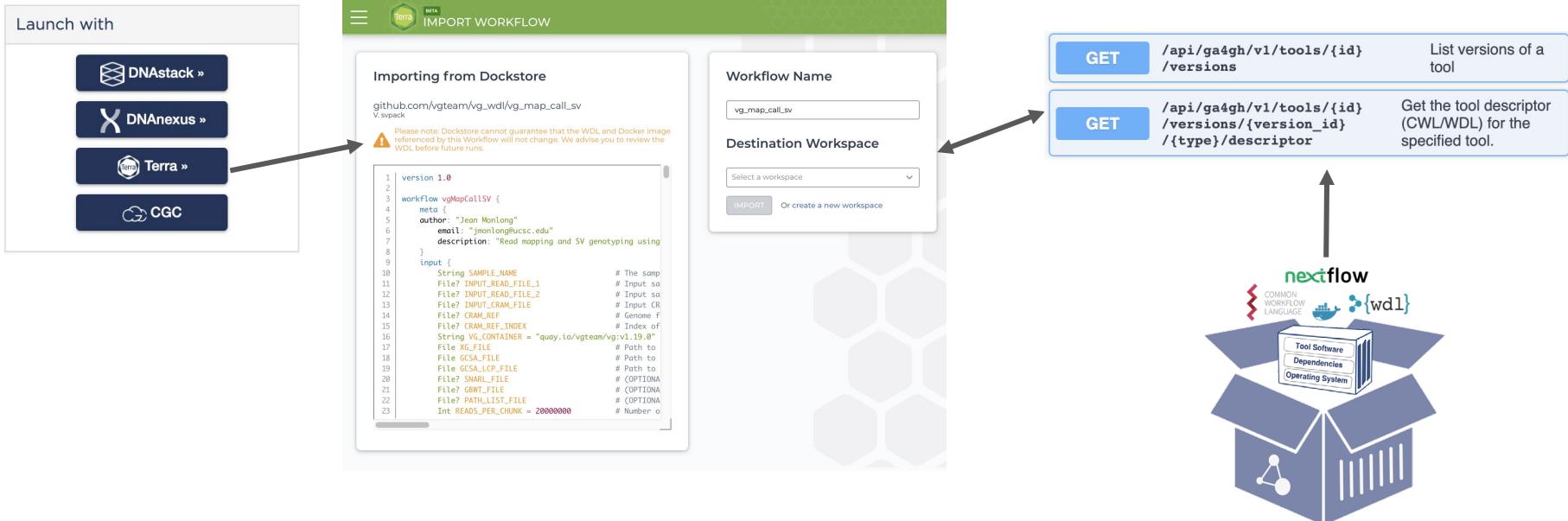


- **GitHub page:** <https://github.com/ga4gh/tool-registry-service-schemas>
- **Latest draft release:** 1.0.0, working on 2.0.0 now

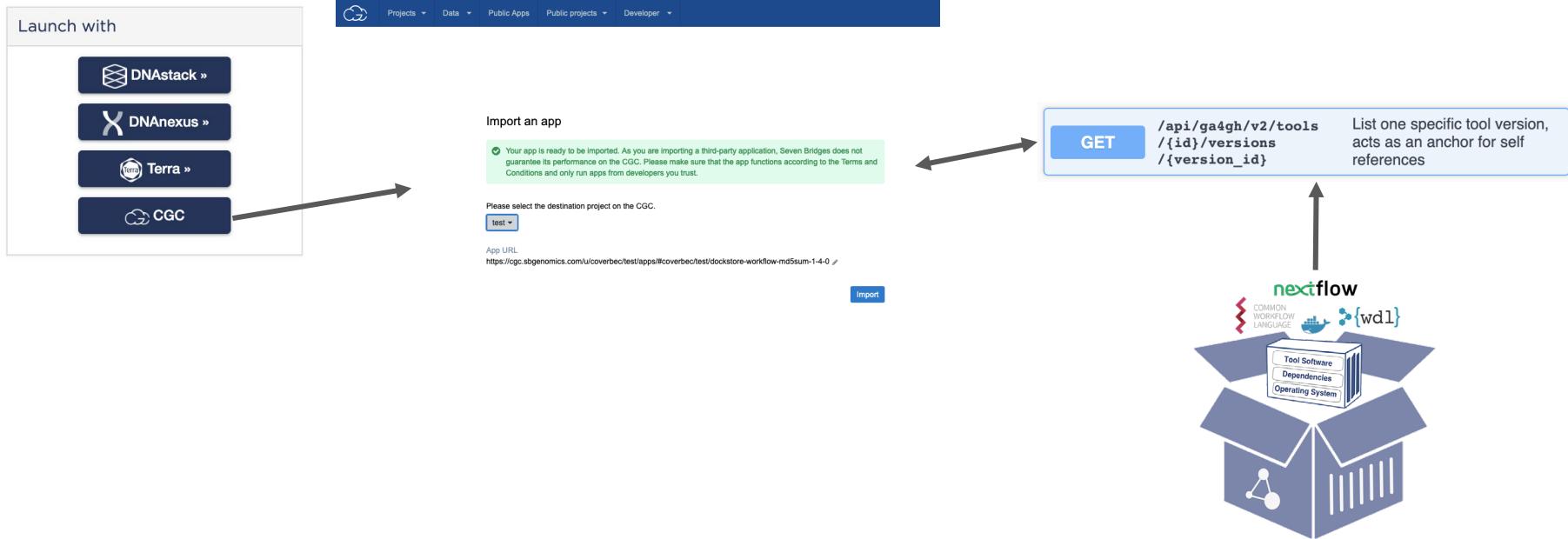


Global Alliance
for Genomics & Health

Importing workflows to Terra using TRS



Importing to Seven Bridges Cancer Genomics Cloud w/ TRS



Documentation and Tutorials

- Example Topics:
 - Getting Started With...(Docker, Nextflow, CWL, WDL)
 - Launching Tools and Workflows
 - Writing checker workflows
 - Developing File Provisioning Plugins
 - Creating Organizations
 - And many more!

<https://docs.dockstore.org/>

Developer Tutorial

Go through the process of creating a tool and registering it on Dockstore.

End User Tutorials

Learn how to use Dockstore from the perspective of a user who runs tools and workflows.

Advanced Tutorials

A collection of articles and tutorials regarding advanced Dockstore topics

Documentation / Advanced Developer / Organizations and Collections

Organizations and Collections

Organizations

Organizations are landing pages for collaborations, institutions, consortiums, companies, etc. that allow users to showcase tools and workflows. This is achieved through the creation of collections, which are groupings of related tools and workflows. The users of an organization do not need to own the tools or workflows in any way; the tools and workflows just have to be published. Collections can be thought of as a playlist on a music streaming service where tools and workflows are analogous to individual songs. They can be shared publicly, and the user does not need to own them.

Creating an organization

To create an organization request, go to the organizations page and select [Create Organization Request](#). Any user can request to create an organization by filling out the following form. For now, the request must be approved by a Dockstore curator in order to be public. Until it is approved, you are still able to edit it, add collections, add members, etc.

Create Organization Request

Fill out the form to send an organization request to a Dockstore curator to approve. Once approved, the organization will be publicly visible.

Name *

OICR

The name of the organization

Display Name *

Ontario Institute for Cancer Research

The display name of the organization

Topic *

OICR is a collaborative, not-for-profit research institute accelerating the development of n

A short description of the organization

Organization website

<https://oicr.on.ca/>

Link to organization website

Location

Next Release (1.9.0)

- 1.9.0 - couple months
 - Workflow registration via GitHub Apps
 - First phase of Galaxy support
 - Continuing performance, security, and usability
 - GA4GH Tool Registry Service TRS v2
 - Workflow lifecycle improvement -- archiving
 - Refine services schema
- 2020 Roadmap Highlights
 - Galaxy - workflows and launch
 - Continuing development and roll out of Apps and Services
 - Linking ORCID accounts
 - More details see full [roadmap](#)

The Dockstore Team



Lincoln Stein

Denis Yuen

Andrew Duncan

Gary Luu

Kathy Tran



Brian O'Connor

Benedict Paten

Cricket Sloan

Charles Overbeck

Walt Shands

David Steinberg

Louise Cabansay

Natalie Perez

Melaina Legaspi

Emily Soth

Andy Chen

Acknowledgements



This work was funded by the Government of Canada through Genome Canada and the Ontario Genomics Institute (OGI-168).



Funded by:



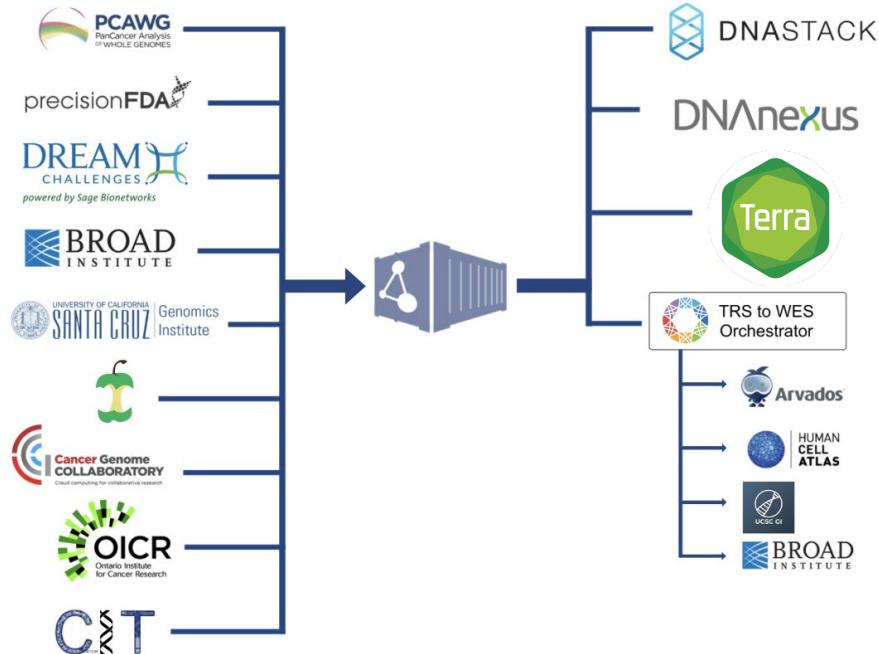
Extra Slides for Q&A

Dockstore Ecosystem

Dockstore is thankful to its many contributors, users, and partners. This community has pulled together a library of over 450 tools and workflows. In the diagram to the right we've highlighted a few select contributors to give a sense of what has been occurring in this space.



Select Contributors and Integrations



TRS V1 vs TRS V2

Breaking Changes

- Conversion of syntax to better support Javascript tooling and better match other GA4GH cloud workstream APIs
- Removal of protobuf support
- Changes to better support other container types (ex Singularity and other Docker alternatives)
- Allow for multiple container images, multiple authors, multiple verification sources for versions of tools

Substantial Changes

- Addition of Nexflow as a descriptor format
- `/tools/{id}/versions/{version_id}/{type}/files` added to describe all available files a tools
- Checker workflows
- Process for auto-generation of OpenAPI3 copy of the schema

More information: <https://ga4gh.github.io/tool-registry-service-schemas/>

Language Support

Feature	CWL	WDL	Nextflow
Dockstore site			
Tool registration	Yes	Yes	No
Workflow registration	Yes	Yes	Yes
Hosted Workflows	Yes	Yes	Yes
DAG Display	Yes (cwl version=>1.0) ^[0]	Yes (wdl version<=draft-2) ^[1]	Limited support
Tool Tab Display	Yes (cwl version=>1.0)	Yes (wdl version<=draft-2)	Yes
Launch-with Platforms	Not yet!	FireCloud (workflows only) ^[2] DNAstack (workflows only) ^[3] DNAnexus (workflows only) Terra (workflows only)	Not yet!
Metadata Display	Yes	Yes (wdl version<=draft-2)	Yes
Dockstore CLI			
Local workflow engines	cwltool, Cromwell	Cromwell ^[4]	Nextflow
File Provisioning In	Local File System HTTP FTP S3 via plugins Data Object Service	Local File System HTTP FTP S3 via plugins Data Object Service	Local File System HTTP FTP S3
Plugins Support	s3 s3cmd icgc-get Data Object Service	s3 s3cmd icgc-get Data Object Service	No
File Provisioning Out	Local File System HTTP FTP S3 via plugins	Local File System	Local File System S3
Notifications	Yes	Yes	No

More info:

<https://docs.dockstore.org/docs/user-tutorials/language-support/>

Ways to Register to Dockstore

My tools

quay.io/lcabansay

Refresh All

My workflows

github.com/abelhj

github.com/CancerCollaboratory

github.com/DataBiosphere

github.com/dockstore

github.com/ga4gh

github.com/Lcabansay

Refresh All

Through linked account
via supported public
APIs

Register Workflow

1 Workflow storage type

2 Create a workflow

You can write your CWL/WDL and store it on Dockstore.org, or you can register a CWL/WDL/NextFlow from an existing third party repository, such as GitHub, Bitbucket, etc.

Use CWL, WDL or Nextflow from GitHub, Bitbucket, etc.

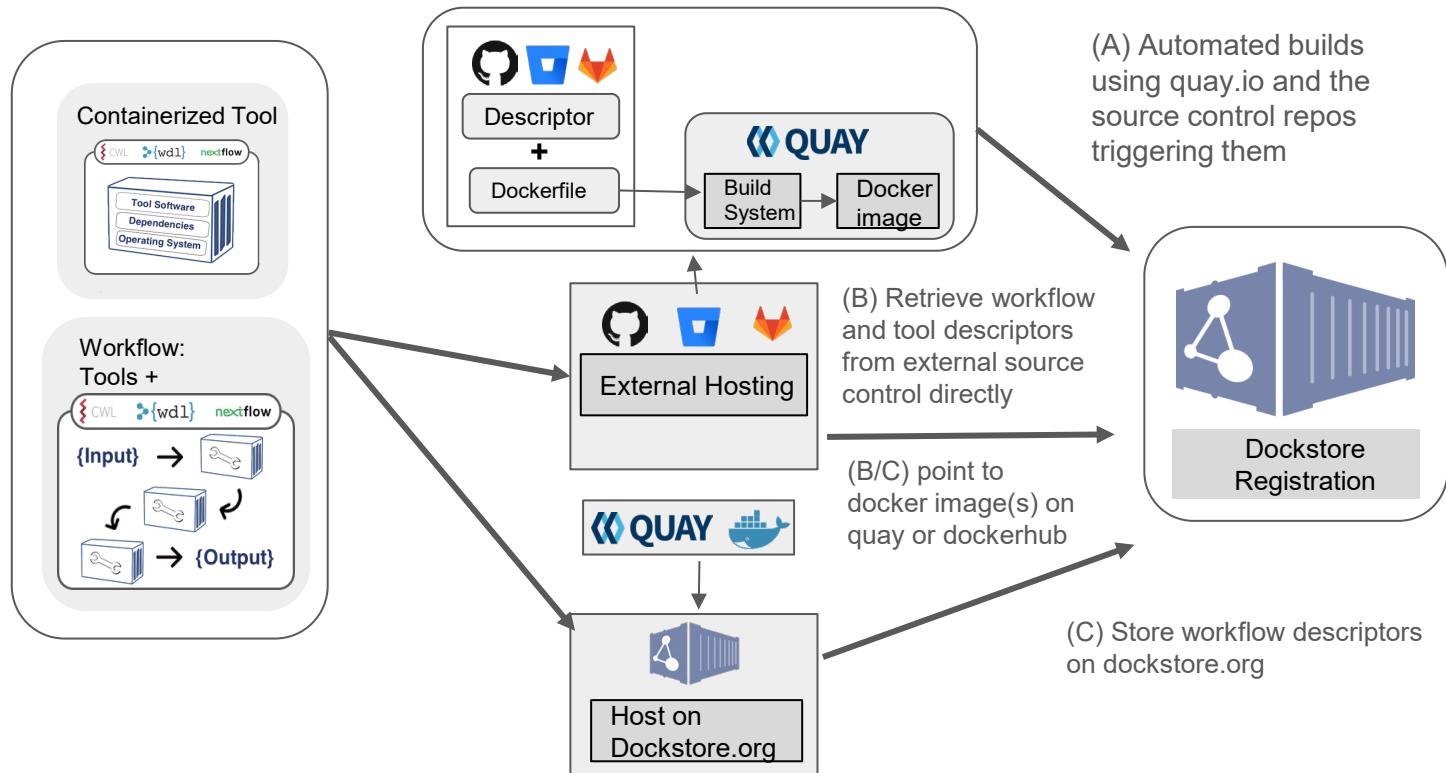
Create and save CWL, WDL, or Nextflow on Dockstore.org

[Close](#)

[Next](#)

OR through manual registration
pointing to external source repo or
directly on Dockstore.org

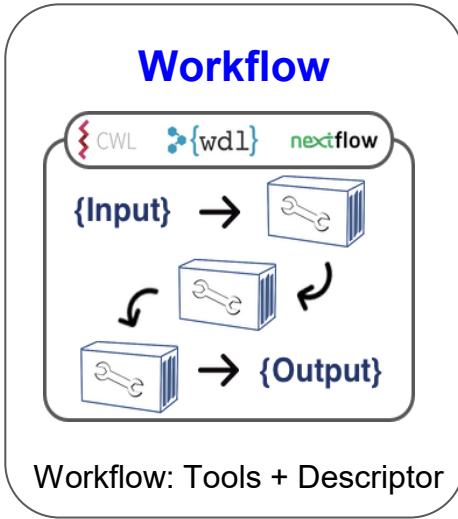
Ways to Register to Dockstore



Content: Tool vs. Workflow

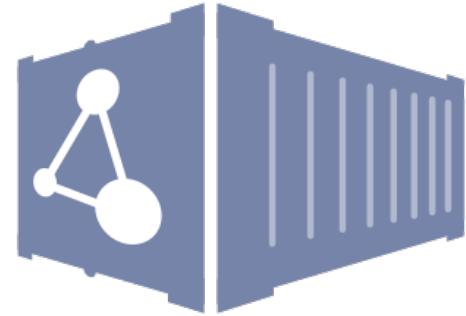


OR



A tool is single container wrapped in a descriptor

A workflow can have multiple containerized tools wrapped in a descriptor

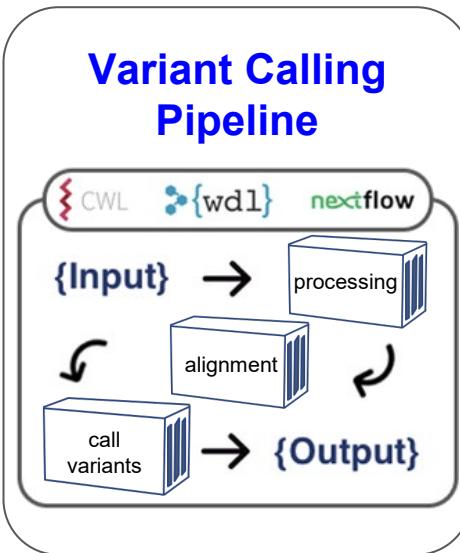


Content: tool vs workflow (example)

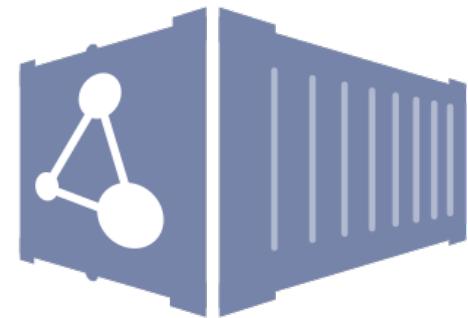
Tool



OR

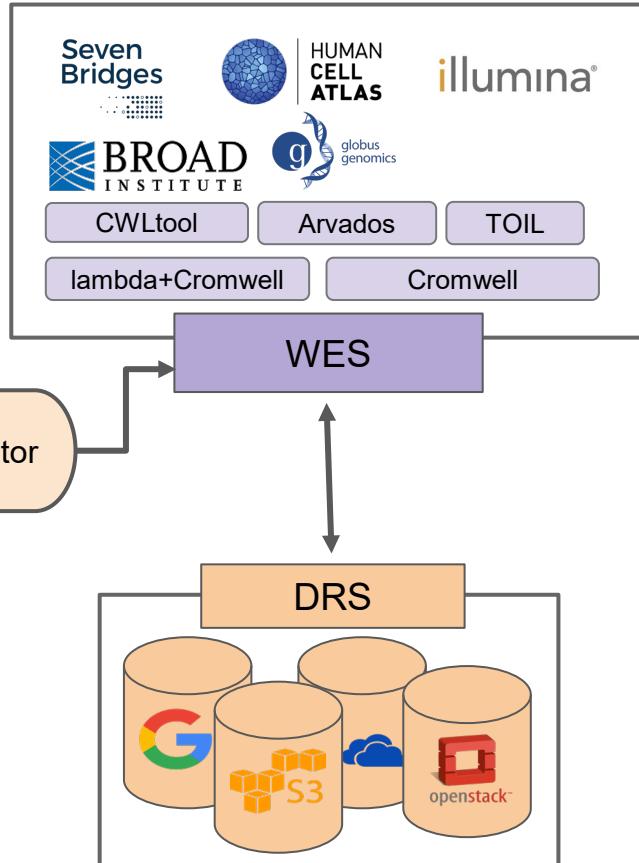
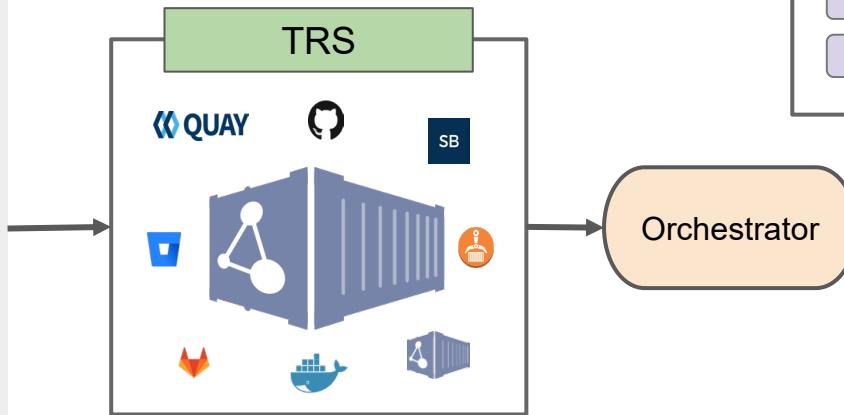
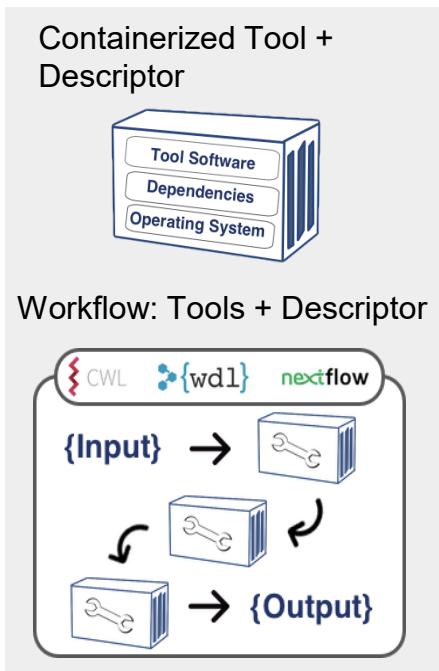


A tool is single container with a descriptor that performs a single task.

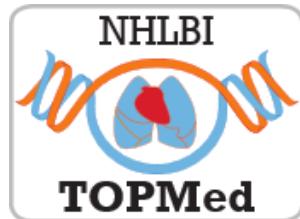


A workflow can have multiple containers wrapped in a descriptor and performs multiple tasks.

Dockstore & GA4GH Cloud Workstream



Collaborations - *The Commons Alliance and Other Projects*



**Da
ta STAGE**

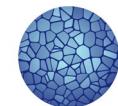
**NHLBI Data
STAGE**



National Human
Genome Research
Institute



**NHGRI
AnVIL**



HUMAN
CELL
ATLAS



And Many Others

Dockstore will serve as an official workflow provider for the Commons Alliance, a common infrastructure collaboration to facilitate combining Data + Cloud Compute