Common Workflow Language

Slides by Cody Receno and Jonathan Dursi

Materials available at
https://github.com/screx/cwl-tutorial

Pipelines can be complicated!

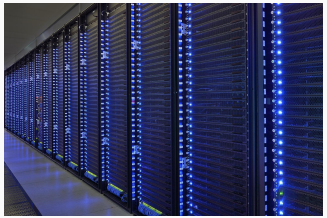
Pipelines are chains of different software tools run sequentially/in parallel that transform the raw reads from NGS into something that can be interpreted.

Pipelines can be hard to understand and modify the more complex they get and optimizing them for different systems may be difficult.

Computing Systems are complicated!

Analysis can be hard to **reproduce** by other groups running the same or similar pipelines.

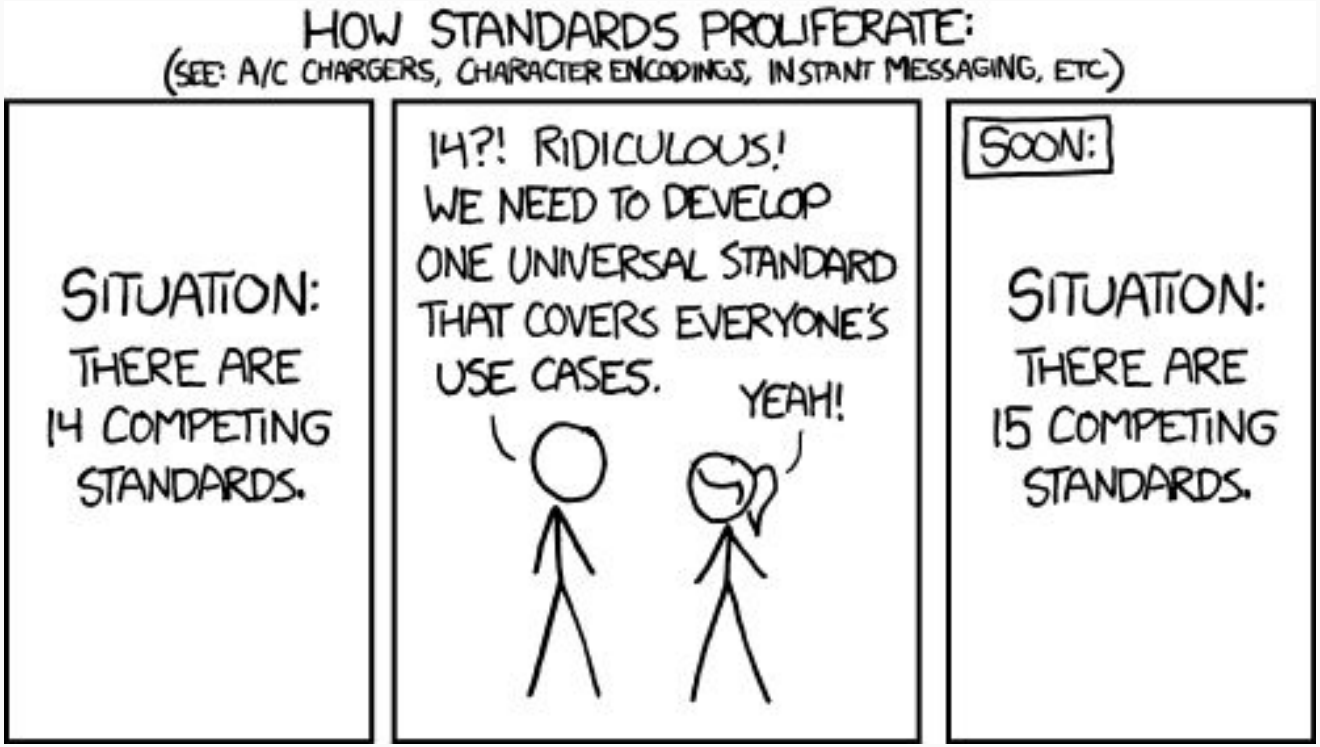
Different systems may have different required system requirements.



Requests and Data Transfer Prices Comparison

What is needed?

- A way to standardize a way of describing pipelines separate from how they are run.
- Be able to run across different computing systems
  - Desktops
  - Clouds
  - Clusters
- A couple of different approaches via CWL and WDL



Competing Standards? Say no more!


CWL

Community Supported!

Slide from Crusoe, M (2016)

Interoperability

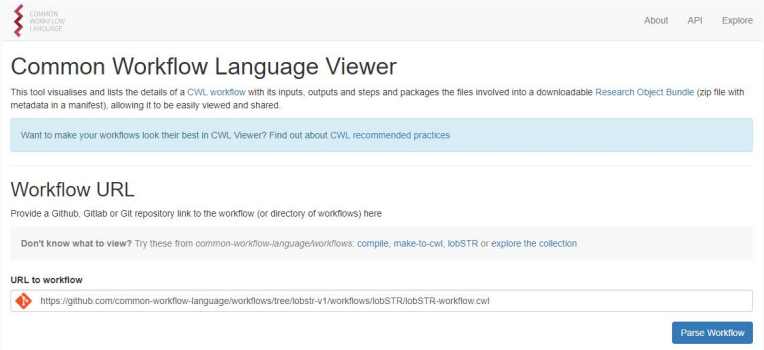
Can easily share and run workflows across different platforms as it would ideally work on different computing systems.

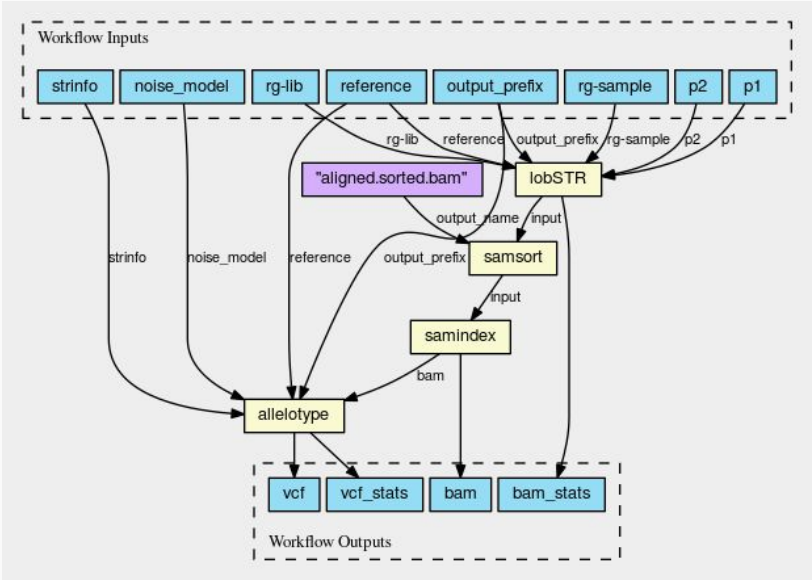
Write workflows at home then send them to the cloud!

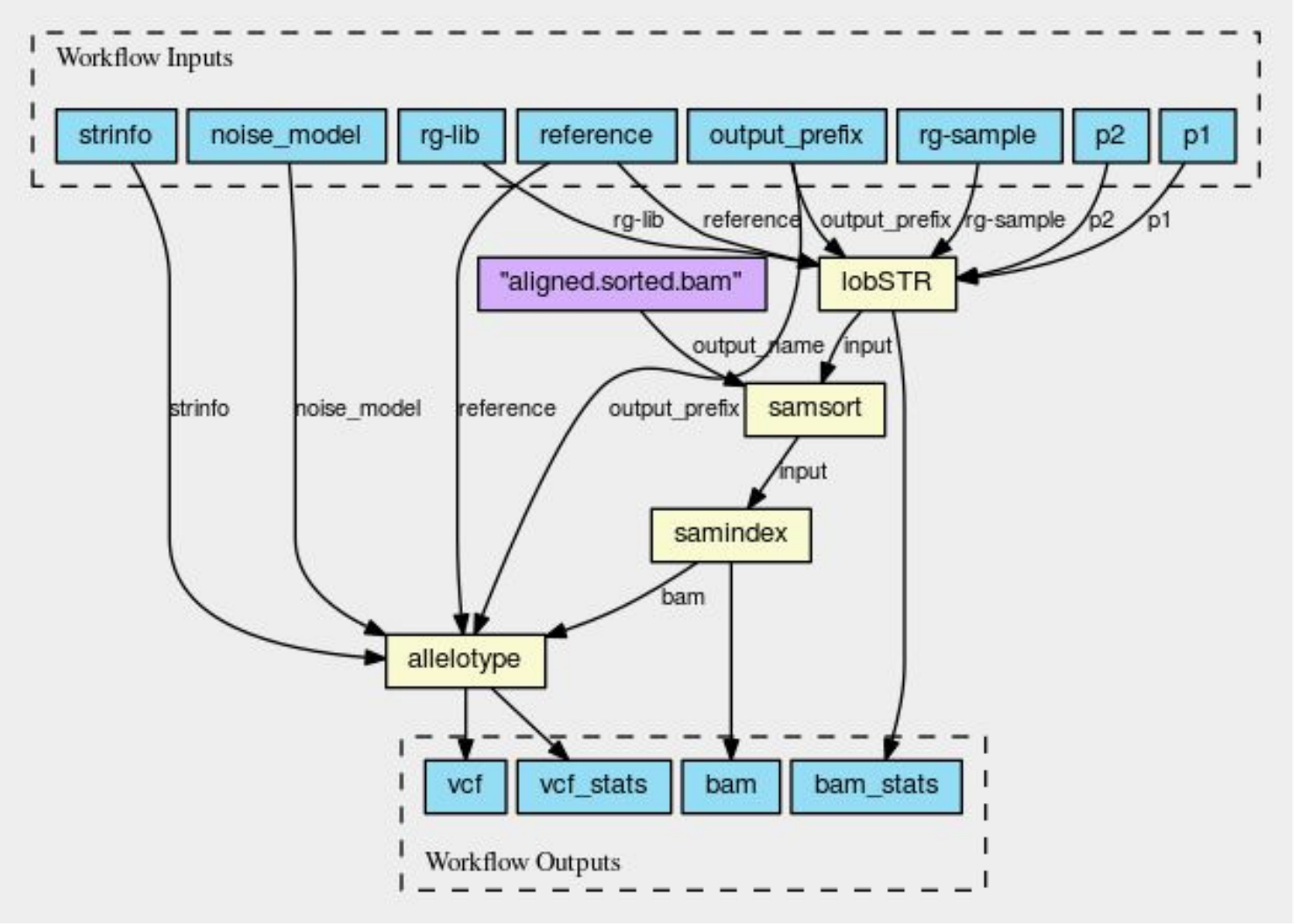
Can reproduce results by sharing the workflows and having other users verify.

Visualizing Workflows

Open source workflows written in cwl that are shared on github can be visualized using view.commonwl.org

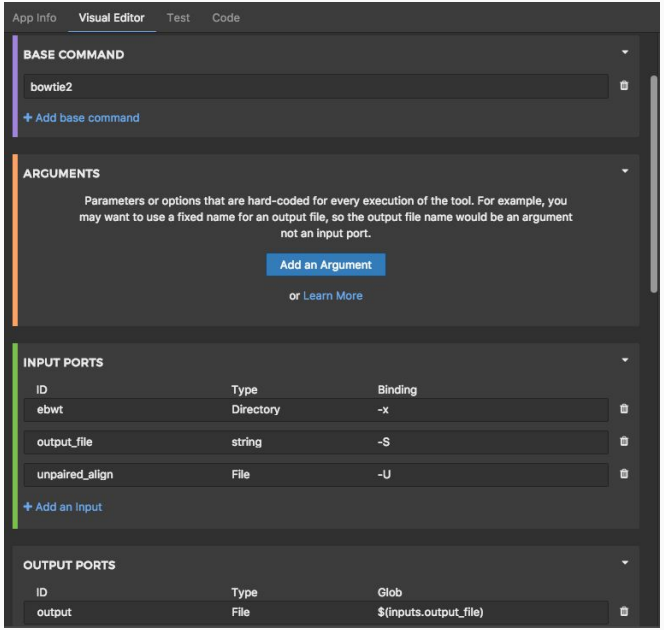


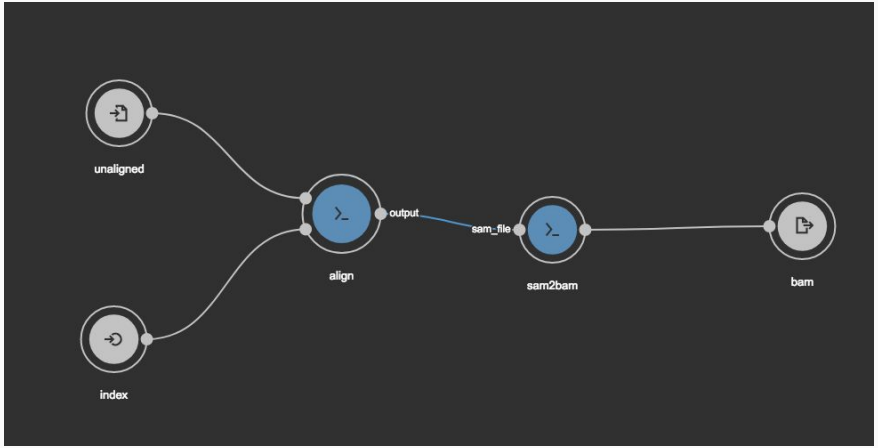




lobstr workflow

Rabix Composer





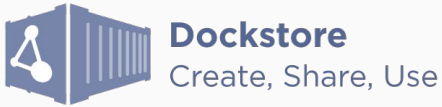
https://github.com/rabix/composer

More Information and tools

commonwl.org - the hub for CWL activity

BCBio - another set of pipelines that can use CWL for many of its workflows

dockstore.org - home to packaged software (containers), and CWL for tools and pipelines



Installation

Open terminal

Login to the hpf server

Use the qlogin command.

Installing requirements:

```
$ source /home/ljdursi/CWL/setup.sh
```

Successfully installed cwl reference runner

```
$ cwltool --help
```

Copy folder over:

```
$ cpcwl
```

Change Directories:

```
$ cd CWL/starter-files
```

Building Workflows: Tools

GOAL: Describe and run various software tools using CWL

Software Tools

Software tools represent the steps/nodes in a workflow and are usually commands that can be used in a terminal.

e.g BWA mem, Picard, grep, echo, etc.

In CWL these tools need to be formally described so they can be incorporated later on into workflows.

Software Tools

Different tools take different arguments in different ways.

Must have a way to write the tools in a standard way

explicitly write the inputs/outputs

bcftools filter -H -i input.vcf “AF<0.1” -o output.vcf

Declarative Language

Easy to understand format

YAML syntax

Standardized sections and field names

Easy to write

EXAMPLE: SAMTOOLS-SORT.CWL

File type & metadata

```
class: CommandLineTool
cwlVersion: v1.0
doc: Sort by chromosomal coordinates
```

Runtime environment

```
hints:
  DockerRequirement:
    dockerPull: quay.io/cancercollaboratory/dockstore-tool-samtools-sort
```

Input parameters

```
inputs:
  aligned_sequences:
    type: File
    format: edam:format_2572  # BAM binary alignment format
    inputBinding:
      position: 1
```

Executable

```
baseCommand: [samtools, sort]
```

Output parameters

```
outputs:
  sorted_aligned_sequences:
    type: stdout
    format: edam:format_2572
```

Linked data support

```
$namespaces: { edam: "http://edamontology.org/" }
$schemas: [ "http://edamontology.org/EDAM_1.15.owl" ]
```


Slide from Crusoe, M (2016)

Building Workflows: Pipelines

GOAL: Describe, run, and modify workflows using CWL.

What is a pipeline?

A pipeline is a series of software tools that are manipulated to work together.

In bioinformatics they are typically used to go from raw sequencing data to something that can be interpreted.

We described each of the nodes earlier, now we want to describe the directions between the nodes.

Workflows

Software tools described previously act as the steps of a workflow and can be used to link the output of one tool as the input of another tool.