ENABLING REPRODUCIBLE IN-SILICO DATA ANALISES WITH NEXTFLOW

Paolo Di Tommaso, CRG

Wellcome Trust Sanger Institute, 1 May 2018, Cambridge

WHO IS THIS CHAP?

@PaoloDiTommaso

Research software engineer

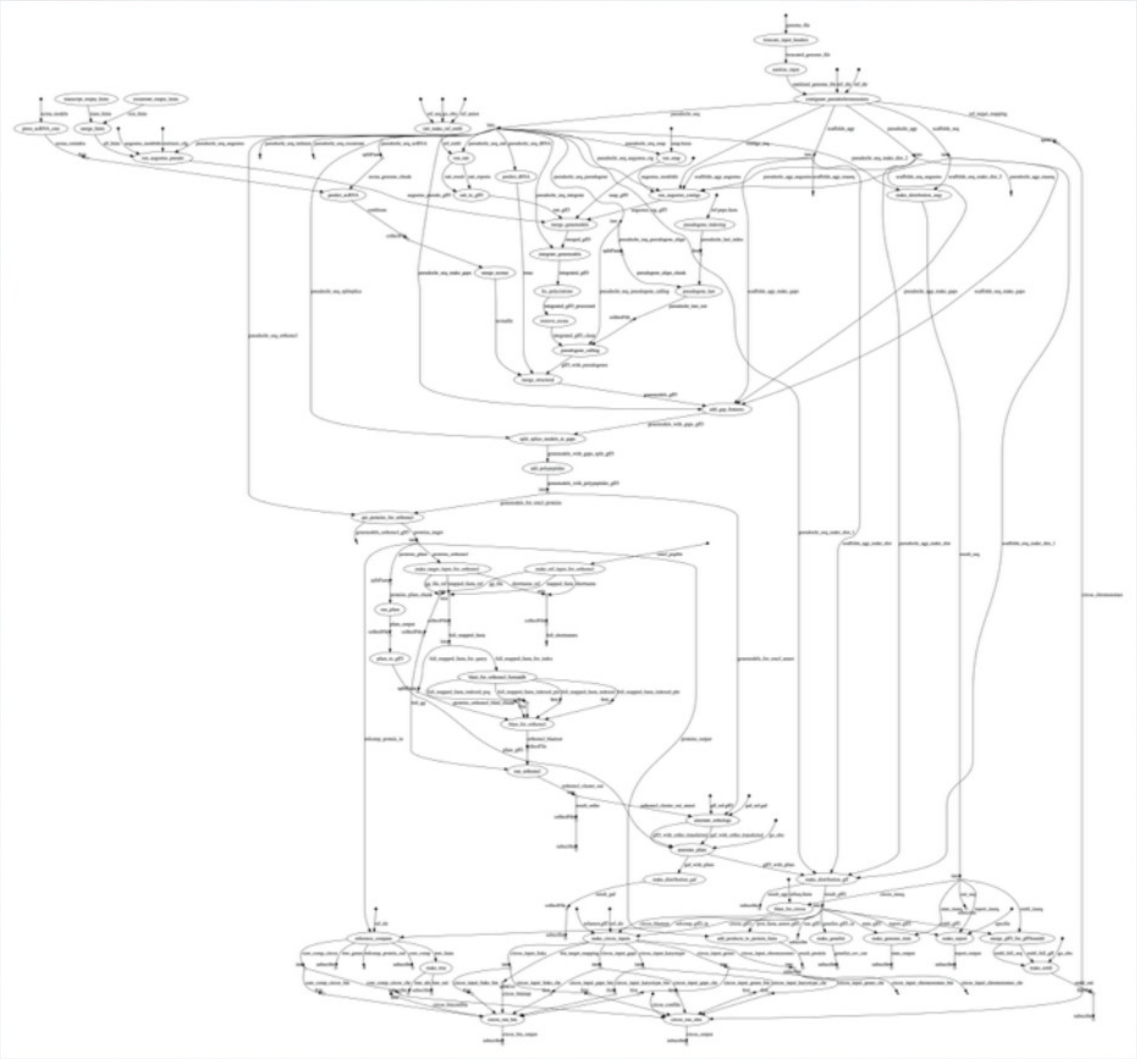
Comparative Bioinformatics, Notredame Lab

Center for Genomic Regulation (CRG)

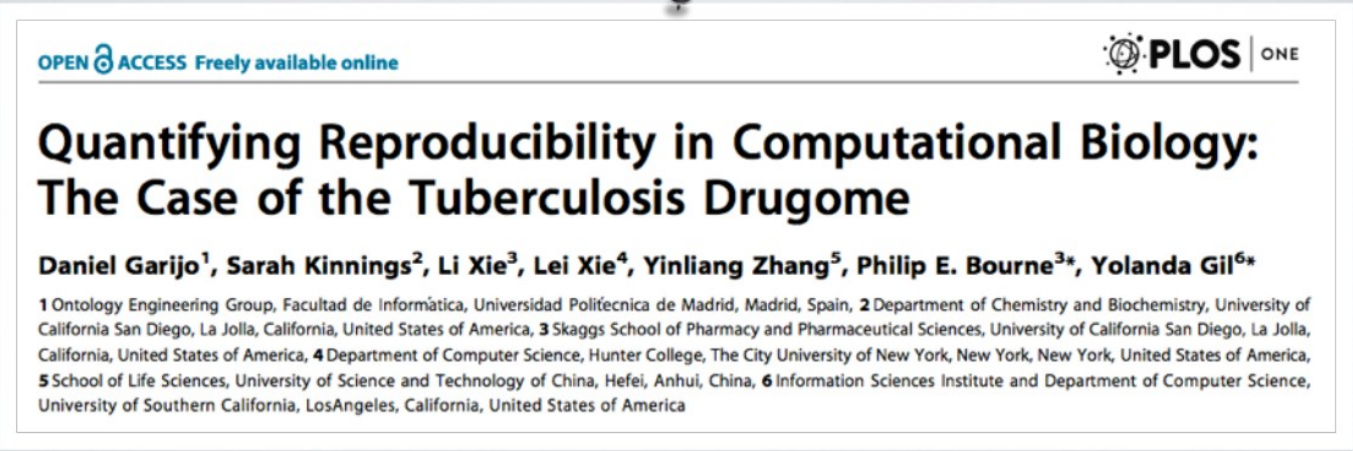
Author of Nextflow project

GENOMIC WORKFLOWS

- Data analysis applications to extract information from (large) genomic datasets
- Embarrassingly parallelisation, can spawn 100s-100k jobs over distributed cluster
- Mash-up of many different tools and scripts
- Complex dependency trees and configuration → very fragile ecosystem



Steinbiss et al., *Companion parassite genome annotation pipeline*, DOI: 10.1093/nar/gkw292



OPEN ACCESS Freely available online

PLOS | ONE

Quantifying Reproducibility in Computational Biology: The Case of the Tuberculosis Drugome

Daniel Garijo[1], Sarah Kinnings[2], Li Xie[3], Lei Xie[4], Yinliang Zhang[5], Philip E. Bourne[3]*, Yolanda Gil[6]*

1 Ontology Engineering Group, Facultad de Informática, Universidad Politécnica de Madrid, Madrid, Spain, **2** Department of Chemistry and Biochemistry, University of California San Diego, La Jolla, California, United States of America, **3** Skaggs School of Pharmacy and Pharmaceutical Sciences, University of California San Diego, La Jolla, California, United States of America, **4** Department of Computer Science, Hunter College, The City University of New York, New York, New York, United States of America, **5** School of Life Sciences, University of Science and Technology of China, Hefei, Anhui, China, **6** Information Sciences Institute and Department of Computer Science, University of Southern California, LosAngeles, California, United States of America

To reproduce the result of a typical computational biology paper requires 280 hours.

≈1.7 months!

THE SAME APPLICATION
DEPLOYED IN
DIFFERENT ENVIRONMENTS
PRODUCES
DIFFERENT RESULTS (!)

Comparison of the Companion pipeline annotation of *Leishmania infantum* genome executed across different platforms *

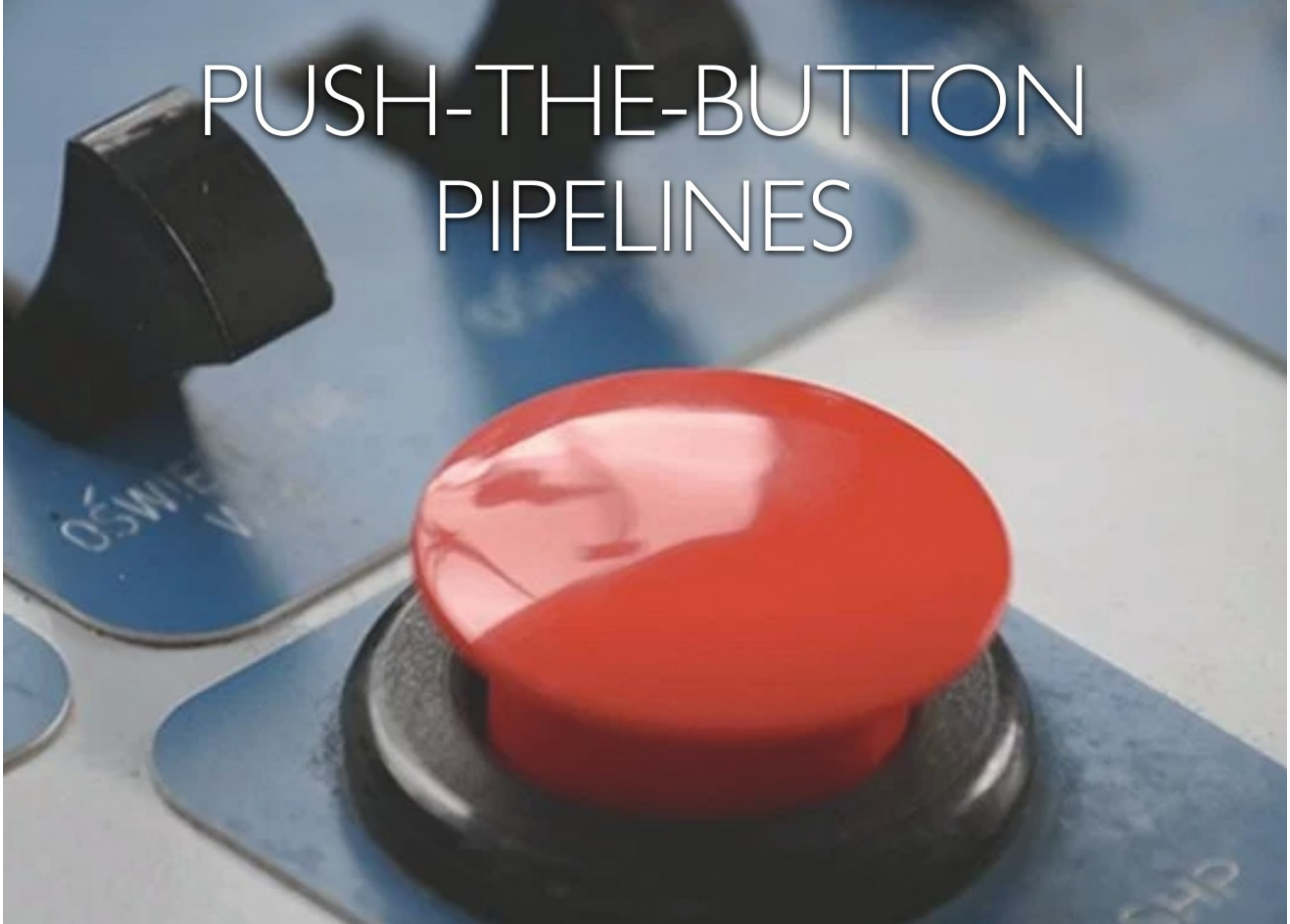
Platform	Amazon Linux	Debian Linux	Mac OSX
Number of chromosomes	36	36	36
Overall length (bp)	32,032,223	32,032,223	32,032,223
Number of genes	7,781	7,783	7,771
Gene density	236.64	236.64	236.32
Number of coding genes	7,580	7,580	7570
Average coding length (bp)	1,764	1,764	1,762
Number of genes with multiple CDS	113	113	111
Number of genes with known function	4,147	4,147	4,142
Number of t-RNAs	88	90	88

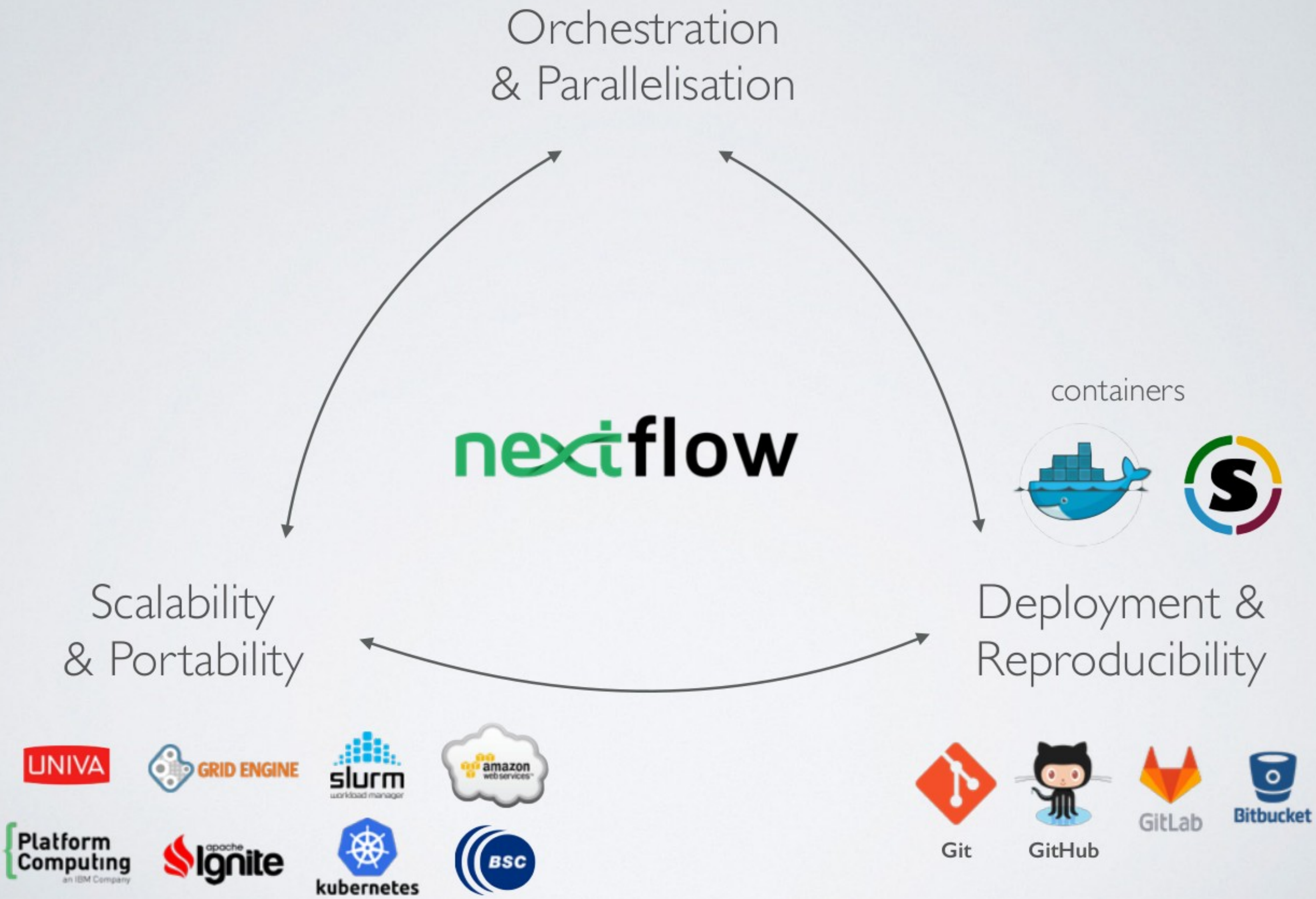
* Di Tommaso P, et al., *Nextflow enables computational reproducibility*, Nature Biotech, 2017

CHALLENGES

- Reproducibility, replicate results over time
- Portability, run across different platforms
- Scalability ie. deploy big distributed workloads
- Usability, streamline execution and deployment of complex workloads ie. remove complexity instead of adding new one
- Consistency ie. track changes and revisions consistently for code, config files and binary dependencies

PUSH-THE-BUTTON PIPELINES





Orchestration & Parallelisation

Scalability & Portability

Deployment & Reproducibility

containers

TASK EXAMPLE

```
bwa mem reference.fa sample.fq \
      | samtools sort -o sample.bam
```

TASK EXAMPLE

```
process align_sample {

    input:
    file 'reference.fa' from genome_ch
    file 'sample.fq' from reads_ch

    output:
    file 'sample.bam' into bam_ch

    script:
    """
    bwa mem reference.fa sample.fq \
          | samtools sort -o sample.bam
     """

}
```

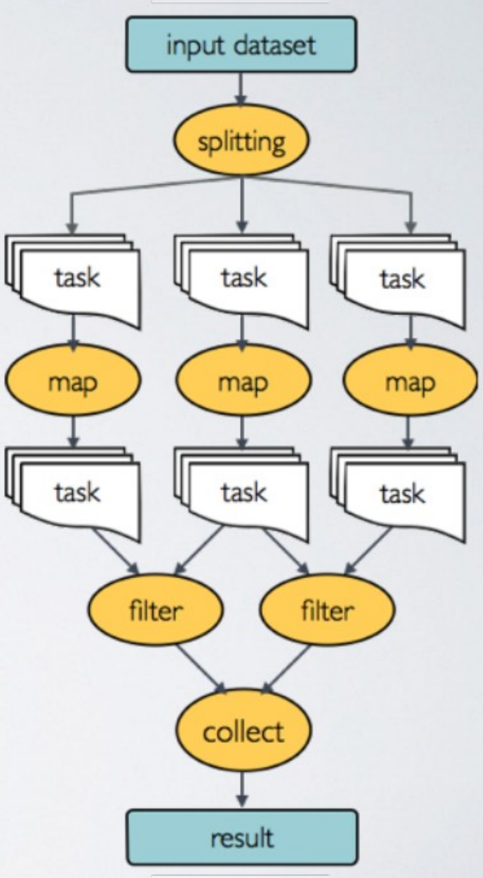
TASKS COMPOSITION

```
process align_sample {

   input:
   file 'reference.fa' from genome_ch
   file 'sample.fq' from reads_ch

   output:
   file 'sample.bam' into bam_ch


   script:
   """
   bwa mem reference.fa sample.fq \
         | samtools sort -o sample.bam
    """

}
```

```
process index_sample {

   input:
   file 'sample.bam' from bam_ch

   output:
   file 'sample.bai' into bai_ch


   script:
   """
   samtools index sample.bam
    """
}
```

DATAFLOW

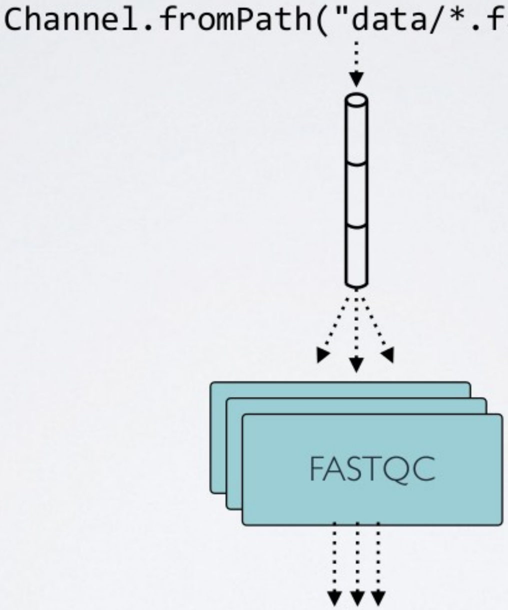
- Declarative computational model for parallel process executions
- Processes wait for data, when an input set is ready the process is executed
- They communicate by using dataflow variables i.e. async FIFO queues called channels
- Parallelisation and tasks dependencies are implicitly defined by process in/out declarations



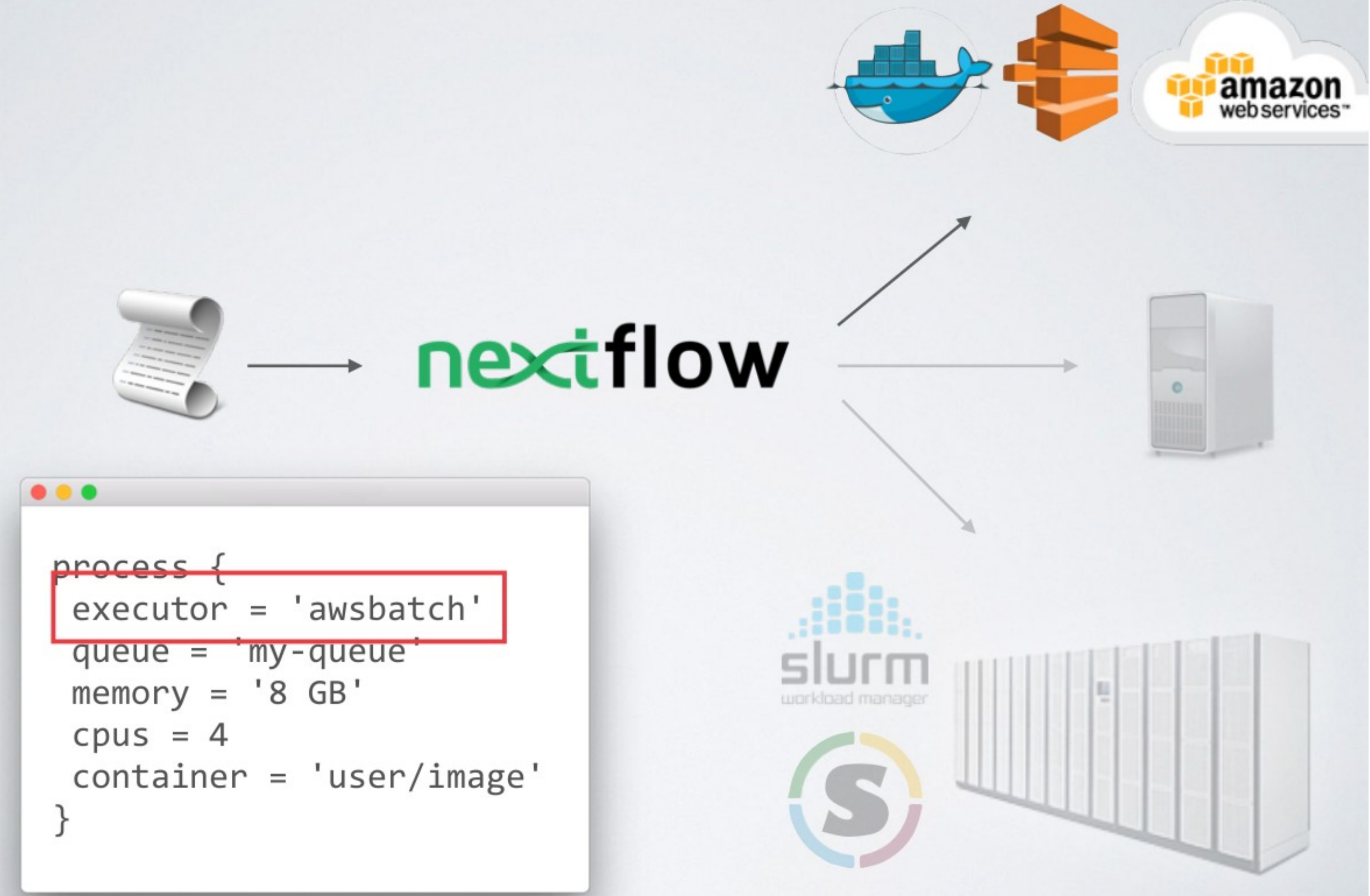
HOW PARALLELISATION WORKS

```
samples_ch = Channel.fromPath('data/*.fastq')


process FASTQC {

   input:
     file reads from samples_ch

   output:
     file 'fastqc_logs' into fastqc_ch

   """
    mkdir fastqc_logs
    fastqc -o fastqc_logs -f fastq -q ${reads}
   """
}
```

IMPLICIT PARALLELISM

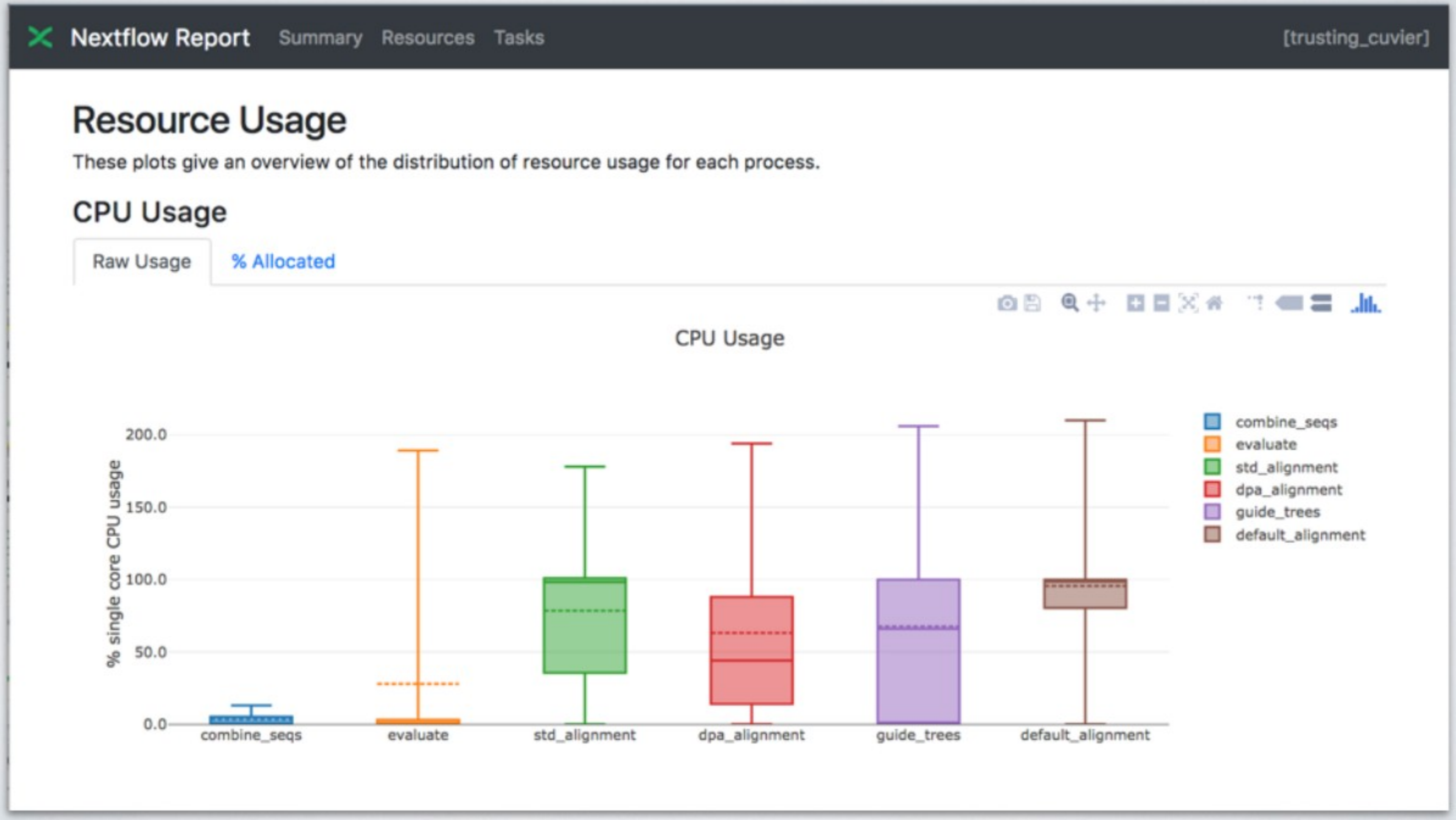


PORTABILITY



```
process {
 executor = 'awsbatch'
 queue = 'my-queue'
 memory = '8 GB'
 cpus = 4
 container = 'user/image'
}
```

EXECUTION REPORT

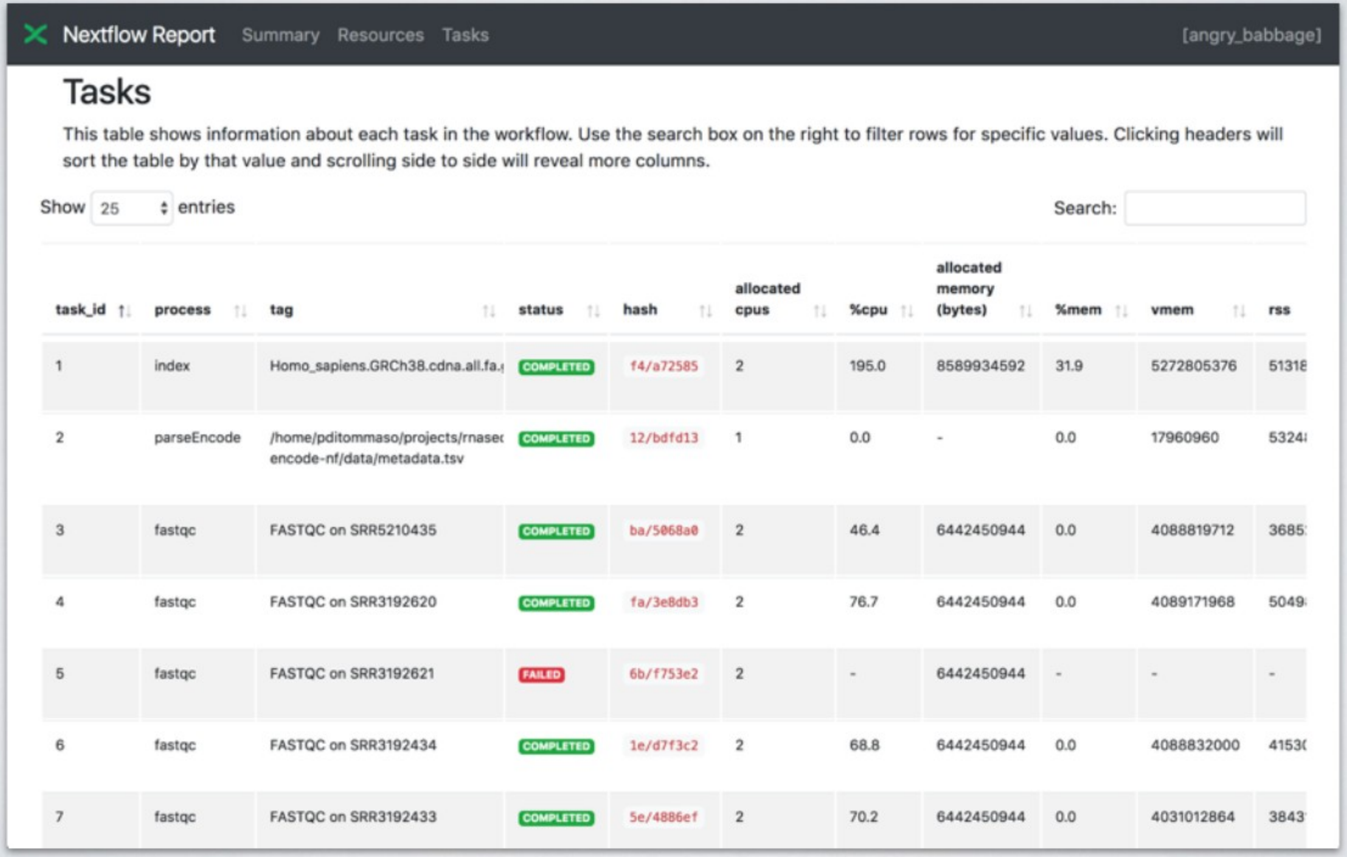


Resource Usage

These plots give an overview of the distribution of resource usage for each process.

CPU Usage

EXECUTION REPORT



Tasks

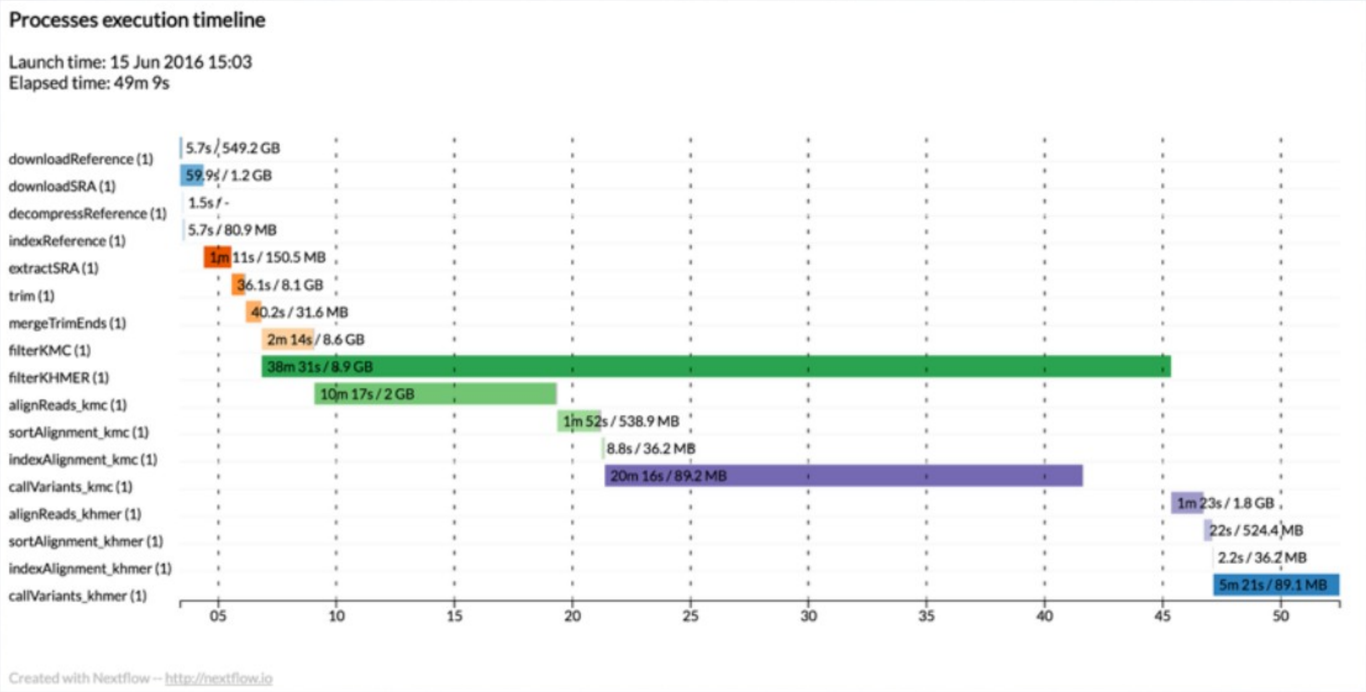
This table shows information about each task in the workflow. Use the search box on the right to filter rows for specific values. Clicking headers will sort the table by that value and scrolling side to side will reveal more columns.

Show 25 entries

Search:

task_id	process	tag	status	hash	allocated cpus	%cpu	allocated memory (bytes)	%mem	vmem	rss
1	index	Homo_sapiens.GRCh38.cdna.all.fa.	COMPLETED	f4/a72585	2	195.0	8589934592	31.9	5272805376	51318
2	parseEncode	/home/pditommaso/projects/rnase encode-nf/data/metadata.tsv	COMPLETED	12/bdfd13	1	0.0	-	0.0	17960960	5324
3	fastqc	FASTQC on SRR5210435	COMPLETED	ba/5068a0	2	46.4	6442450944	0.0	4088819712	3685
4	fastqc	FASTQC on SRR3192620	COMPLETED	fa/3e8db3	2	76.7	6442450944	0.0	4089171968	5049
5	fastqc	FASTQC on SRR3192621	FAILED	6b/f753e2	2	-	6442450944	-	-	-
6	fastqc	FASTQC on SRR3192434	COMPLETED	1e/d7f3c2	2	68.8	6442450944	0.0	4088832000	4153
7	fastqc	FASTQC on SRR3192433	COMPLETED	5e/4886ef	2	70.2	6442450944	0.0	4031012864	3843

EXECUTION TIMELINE

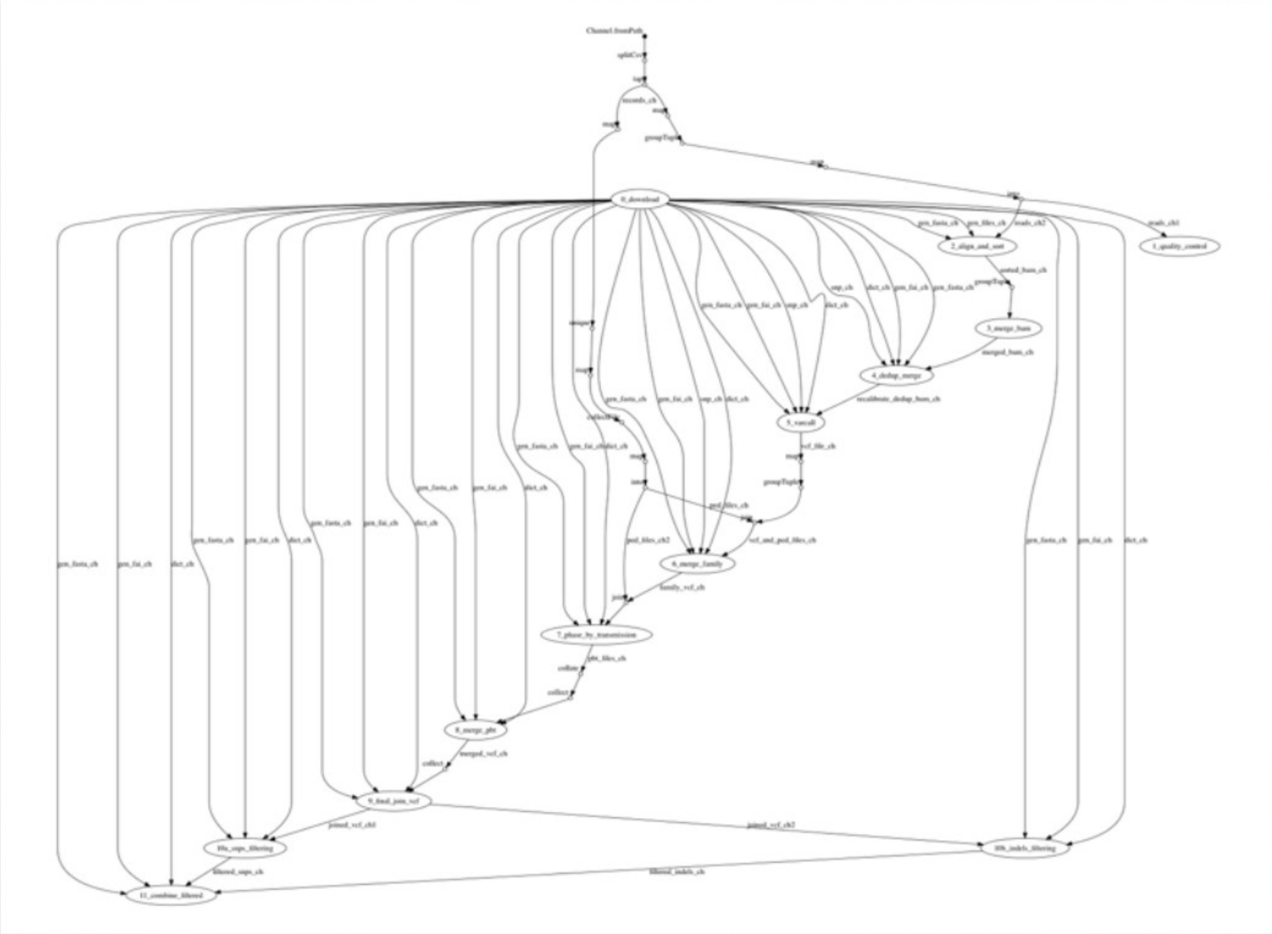


Processes execution timeline

Launch time: 15 Jun 2016 15:03
Elapsed time: 49m 9s

Created with Nextflow -- http://nextflow.io

DAG VISUALISATION



WHO IS USING NEXTFLOW?

UPPSALA UNIVERSITET

Karolinska Institutet

NATIONAL MARROW DONOR PROGRAM®

BERKELEY LAB

JGI JOINT GENOME INSTITUTE DEPARTMENT OF ENERGY

International Agency Research on Cancer

World Health Organization

SciLifeLab

NIH

ILLINOIS UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

UT Southwestern Medical Center

UCSF

UiO: University of Oslo

A*STAR Genome Institute of Singapore

Weill Cornell Medical College

Institut Pasteur

UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Curtin University

QBiC

CRG Centre for Genomic Regulation

bina

wellcome trust sanger institute

gritstone ONCOLOGY

BENAROYA RESEARCH INSTITUTE Virginia Mason

enterome bioscience

SYNTHETIC GENOMICS

Roche

nf-core

- Community effort to collect production ready analysis pipelines built with Nextflow
- Initially supported by SciLifeLab, QBiC and A*Star Genome Institute Singapore
- https://nf-core.github.io

Phil
Ewels

Alexander
Peltzer

Andreas
Wilm