INTRODUCTION TO NEXTFLOW

Paolo Di Tommaso, CRG
NETTAB workshop - Roma

October 25th, 2016

nextflow

CRG Centre for Genomic Regulation

@PaoloDiTommaso
Research software engineer
Comparative Bioinformatics,
Notredame Lab
Center for Genomic Regulation (CRG)

RATIONALE

- Fast applications prototyping
- Simplified model for parallel task executions
- Make pipelines portable and scalable across cluster and cloud platforms
- Enable reproducibility

NEXTFLOW KEY FEATURES

- Custom domain specific language (DSL)
- Dataflow parallel programming paradigm
- Built-in support for different cluster platforms
- Tasks isolation with Docker containers
- Lightweight, command line oriented, no GUI

PROCESS DEFINITION

```
process foo {

    input:
    val str from 'Hello'

    output:
    file 'my_file' into result

    script:
    """
    echo $str world! > my_file
    """
}
```

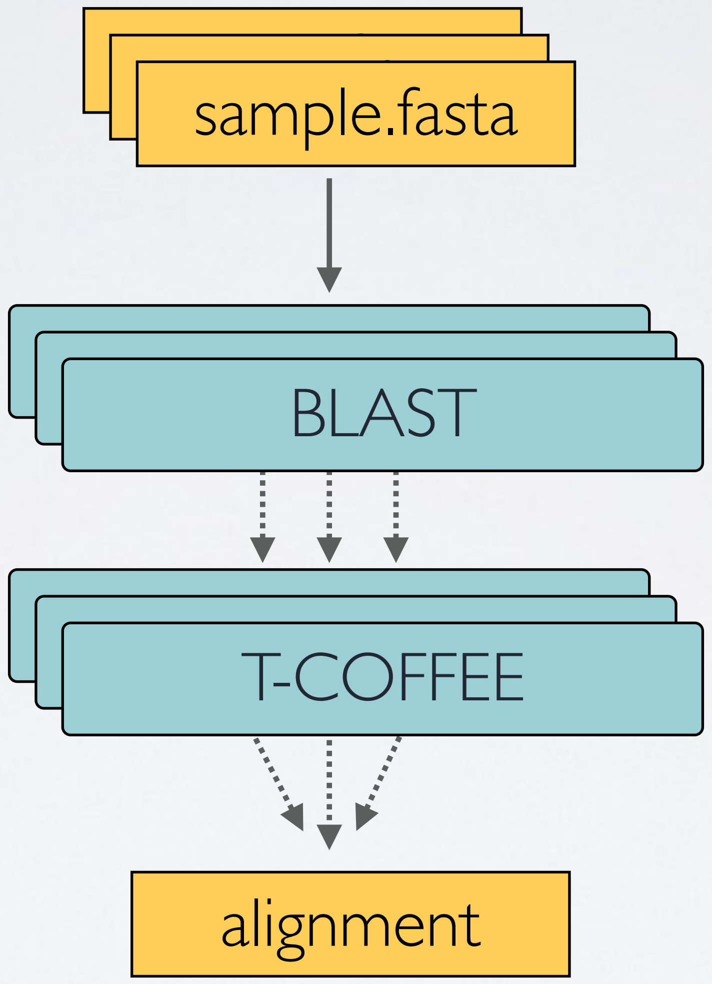
WHAT A SCRIPT LOOKS LIKE

```
sequences = Channel.fromPath("/data/sample.fasta")

process blast {
    input:
    file 'in.fasta' from sequences
    output:
    file 'out.txt' into blast_result

    """
    blastp -query in.fasta -outfmt 6 | cut -f 2 | \
    blastdbcmd -entry_batch - > out.txt
    """
}

process align {
  input:
  file all_seqs from blast_result
  output:
  file align_result

  """
  t_coffee $all_seqs 2>&- | tee align_result
  """
}

blast_result.collectFile(name: 'final_alignment')
```

IMPLICIT PARALLELISM

```
sequences = Channel.fromPath("/data/*.fasta")

process blast {
    input:
    file 'in.fasta' from sequences
    output:
    file 'out.txt' into blast_result

    """
    blastp -query in.fasta -outfmt 6 | cut -f 2 | \
    blastdbcmd -entry_batch - > out.txt
    """
}

process align {
  input:
  file all_seqs from blast_result
  output:
  file align_result

  """
  t_coffee $all_seqs 2>&- | tee align_result
  """
}

blast_result.collectFile(name: 'final_alignment')
```

IMPLICIT PARALLELISM



BENEFITS

- High-level declarative parallelisation abstraction
- Portable across different platforms
- Isolates task dependencies with Docker containers

CONFIGURATION FILE

```
process {
  container = 'your/image:latest'
  executor = 'sge'
  queue = 'cn-el6'
  memory = '10GB'
  cpus = 8
  time = '2h'
}
```

SUPPORTED PLATFORMS

UNIVA

GRID ENGINE

Platform Computing
an IBM Company

slurm
workload manager

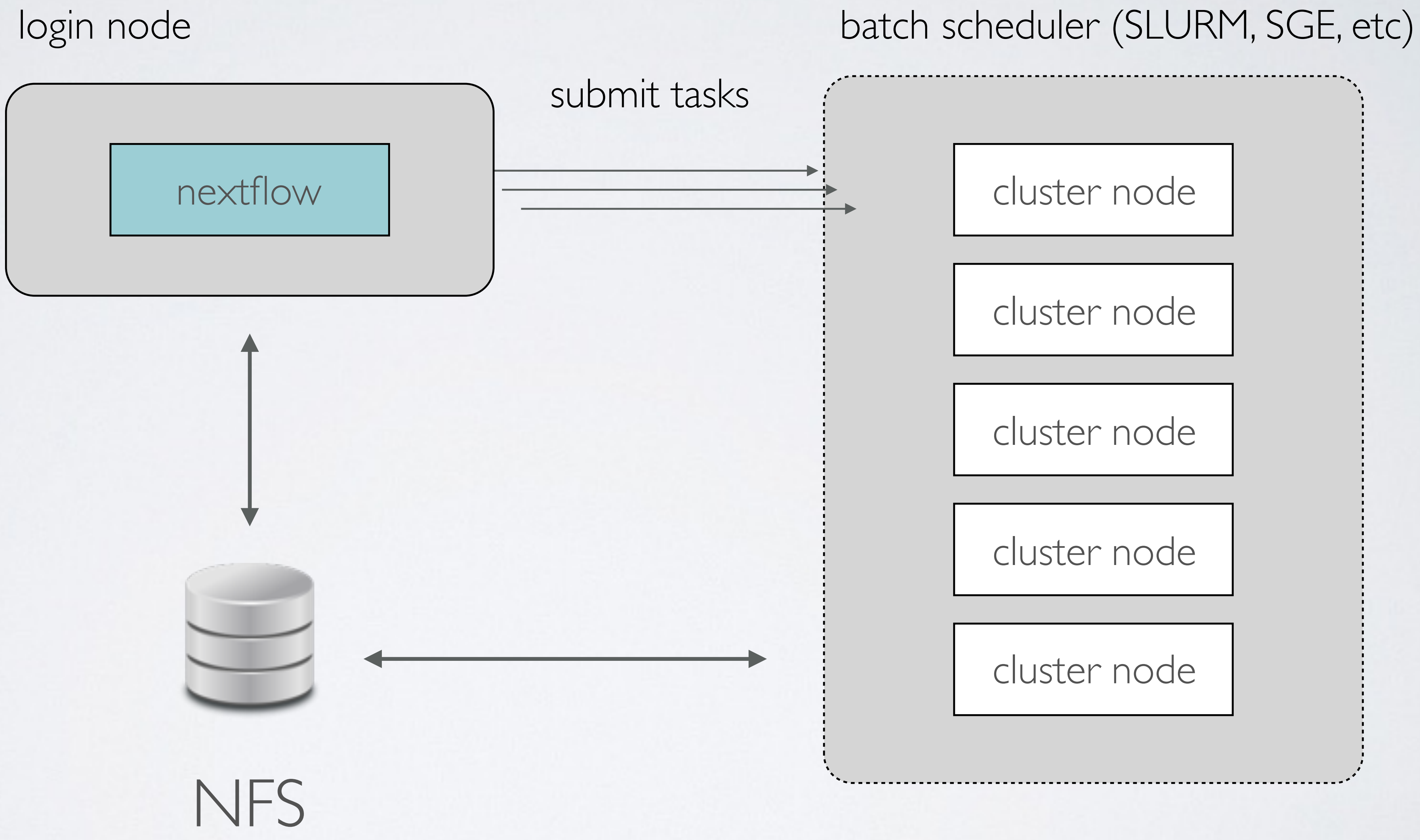
PBS Works™

kubernetes

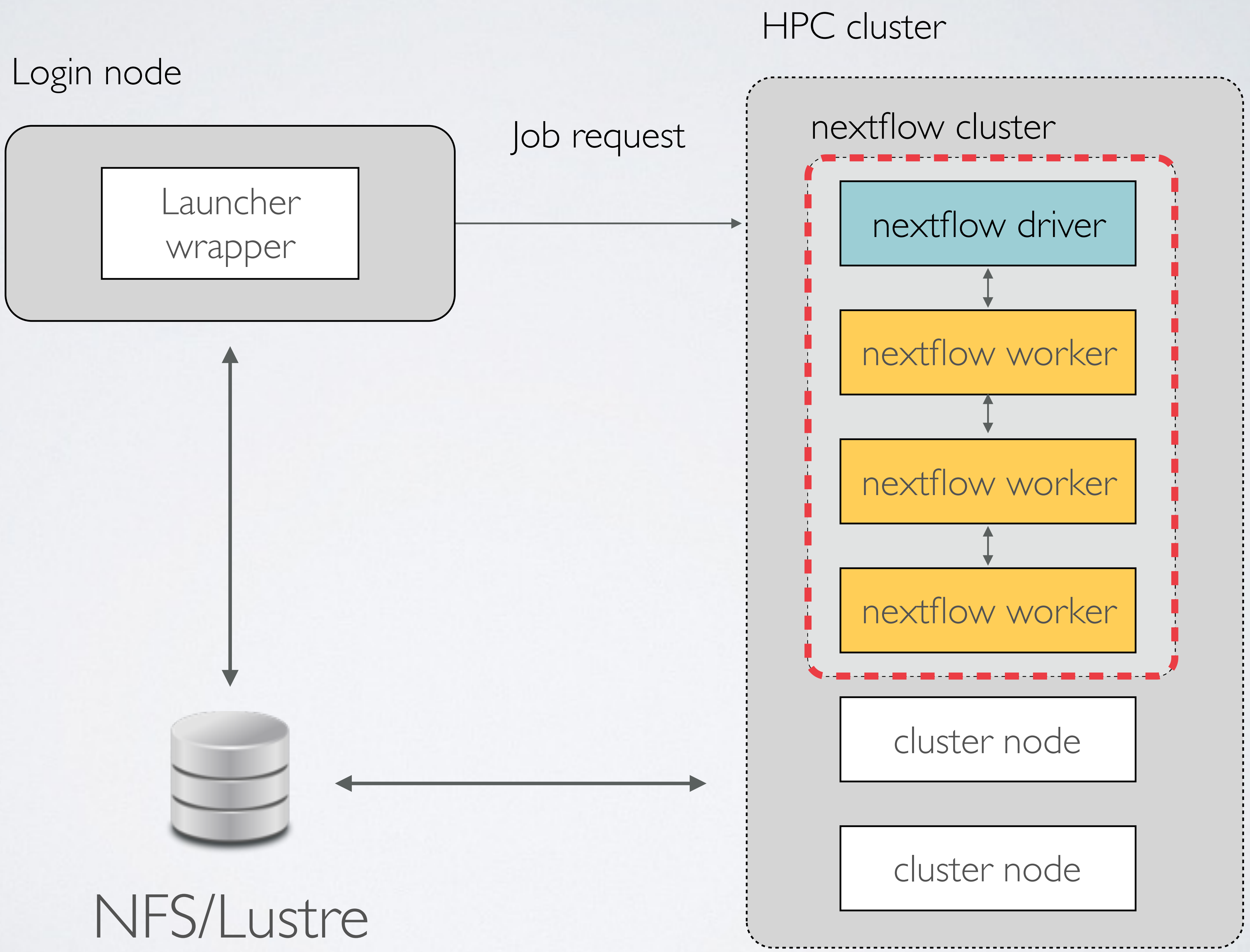
HTCondor
High Throughput Computing

amazon
web services™

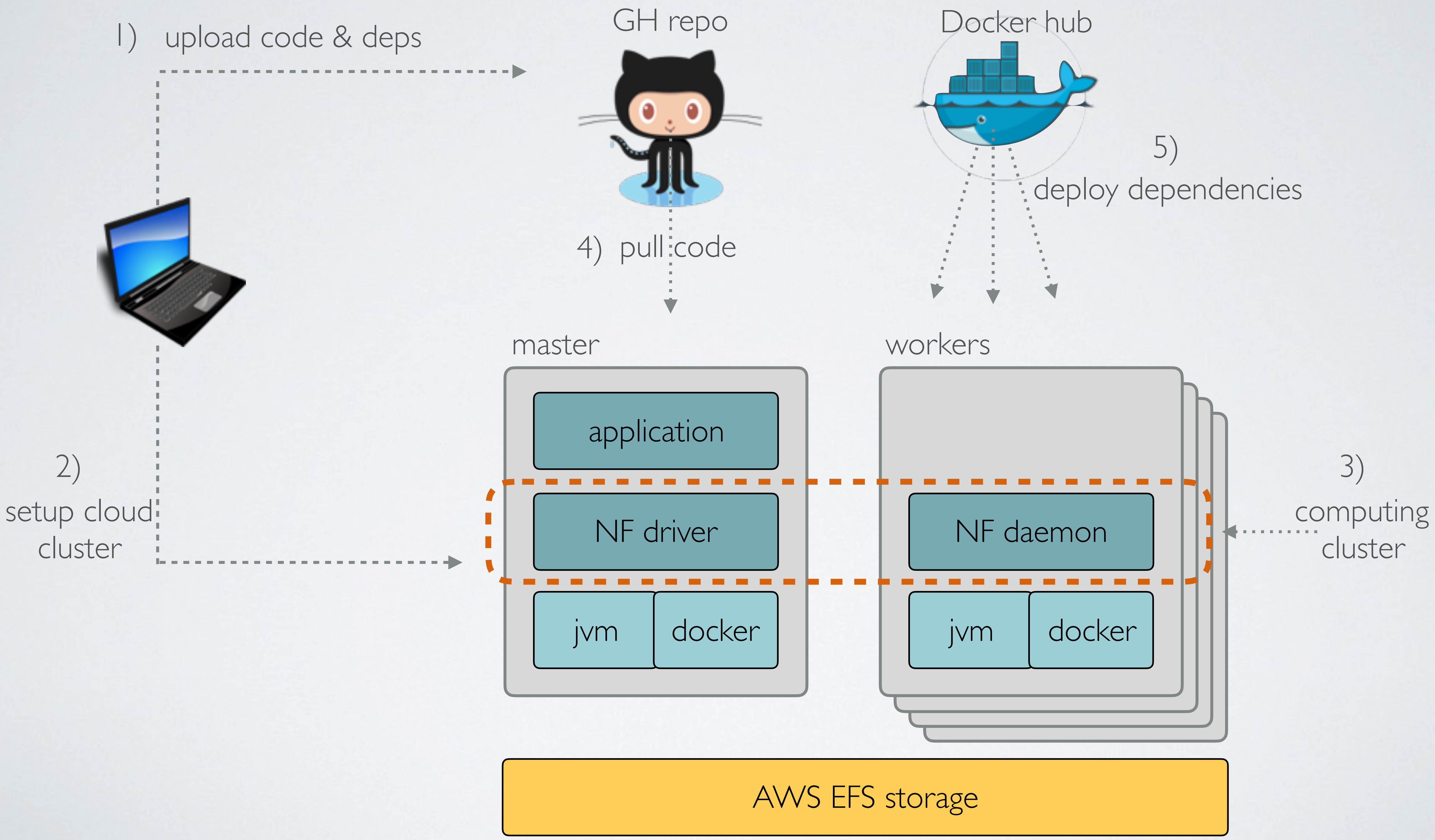
CLUSTER EXECUTION



DISTRIBUTED EXECUTION



CLOUD DEPLOYMENT



ELASTIC CLUSTER

- Native cloud scheduler supporting auto-scaling
- Instances are added on workload pressure
- Instances are terminated when idle
- Support for EC2 spot instances

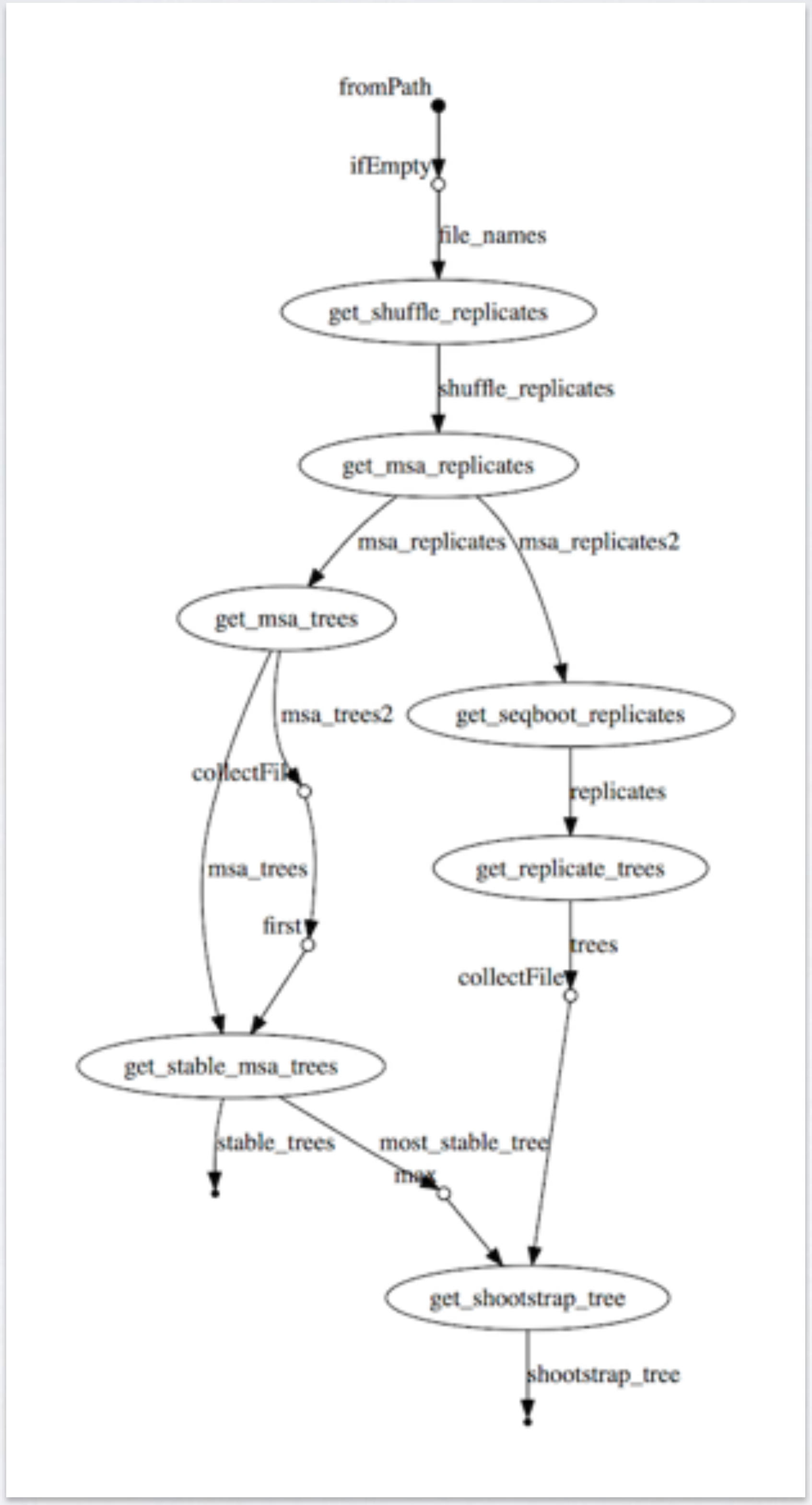
ERROR HANDLING

- Continuous task check-points
- Resume from the last successfully executed step
- Multiple error handling strategies: stop, ignore, retry
- Update resources requirement on failure

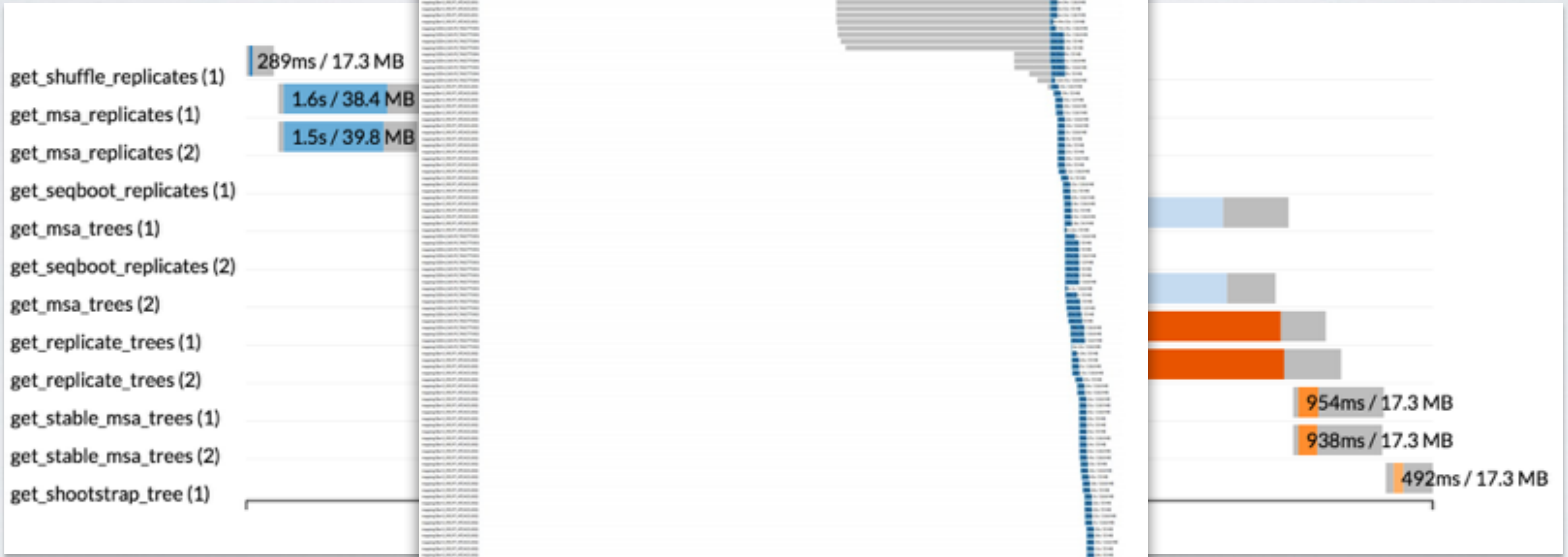
TRACE & VISUALISATION

- Trace tasks runtime metrics such as cpus, memory used, real-time, wall-time, etc.
- Create provenance report including tasks information such as name, container, script executed, work directory, etc.

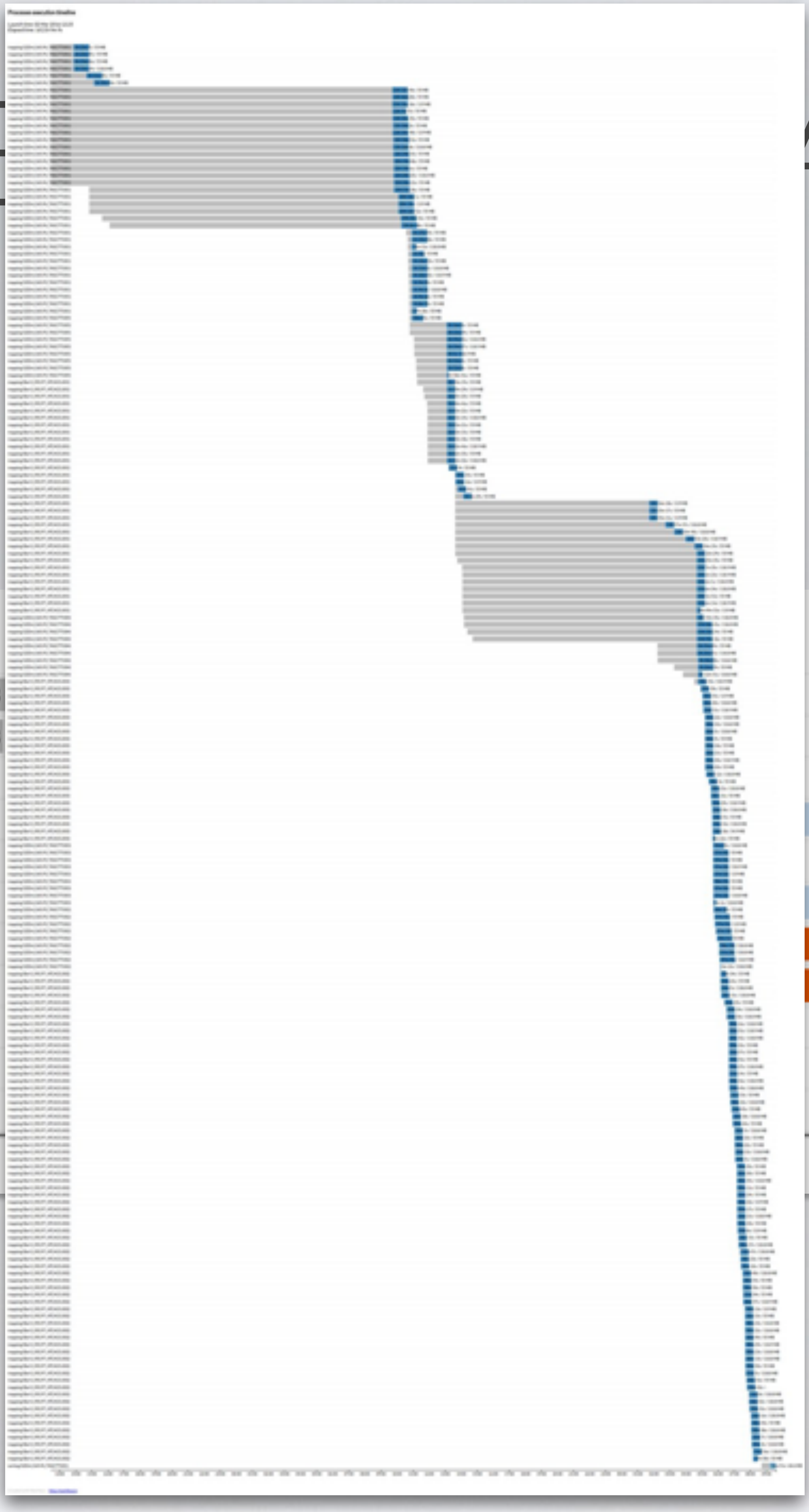
TRACE & VISUALISATION



TRACE ATION



get_shuffle_replicates (1)
289ms / 17.3 MB
get_msa_replicates (1)
1.6s / 38.4 MB
get_msa_replicates (2)
1.5s / 39.8 MB
get_seqboot_replicates (1)
get_msa_trees (1)
get_seqboot_replicates (2)
get_msa_trees (2)
get_replicate_trees (1)
get_replicate_trees (2)
get_stable_msa_trees (1)
954ms / 17.3 MB
get_stable_msa_trees (2)
938ms / 17.3 MB
get_shootstrap_tree (1)
492ms / 17.3 MB



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

ILLINOIS UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

UTSouthwestern Medical Center

UCSF

UiO : University of Oslo

Universität Bielefeld

Weill Cornell Medical College

Institut Pasteur

UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Colorado State University

Curtin University

CRG Centre for Genomic Regulation

bina

wellcome trust sanger institute

PAWSEY supercomputing centre

BENAROYA RESEARCH INSTITUTE Virginia Mason

SYNTHETIC GENOMICS

Roche

DEMO

Implement a proof of concept of a RNA-Seq pipeline which:

1. index a genome file
2. map read pairs against the genome
3. perform quantification

https://github.com/nextflow-io/rna-demo

INSTALLATION

```
curl get.nextflow.io | bash
```

$\downarrow$

```
nextflow
```

* It requires: Unix-like OS (Linux, OSX, etc) and Java 7/8

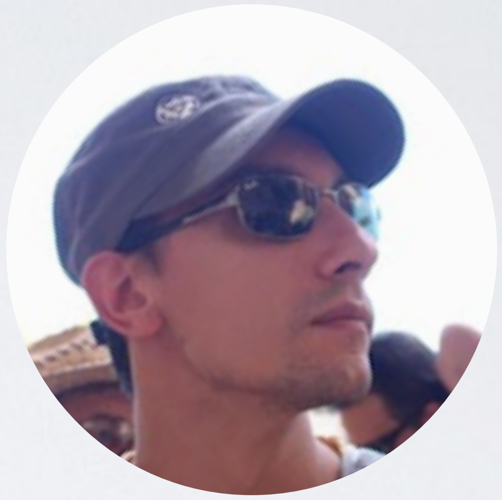
HANDS-ON

HIGHLIGHTS

- Lightweight, command line tool
- Functional-reactive programming model
- Built-in support for multiple execution platforms (clusters, HPC, cloud)
- Integrates community standards such as Docker and Git/GitHub to enable reproducible deployments

ACKNOWLEDGMENT

Evan Floden, CRG



Emilio Palumbo, CRG

Maria Chatzou, CRG

Cedric Notredame, CRG

THANK YOU

LINKS

project home

http://nextflow.io

gitter

https://gitter.im/nextflow-io/nextflow

demo

https://github.com/nextflow-io/demo
https://goo.gl/1F1Ac4