

Reproducible bioinformatics for everyone:

# Nextflow & nf-core



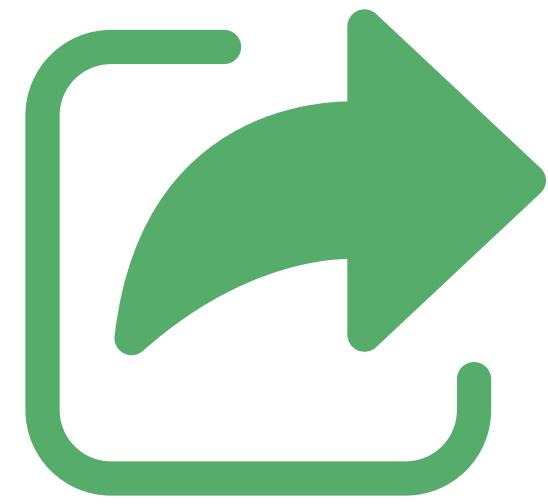
# Reproducible bioinformatics



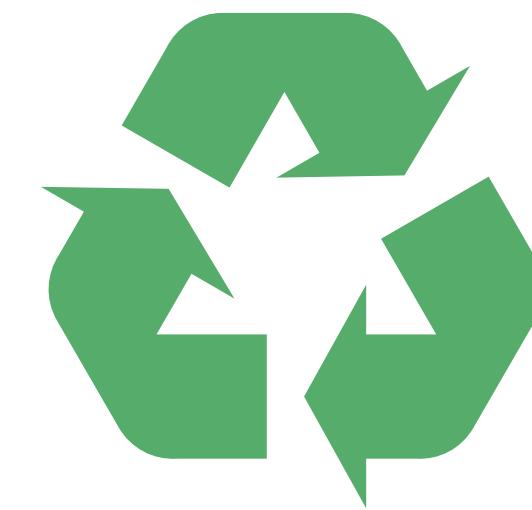
# Reproducible bioinformatics



Get the same results  
every time



Other people also  
get the same results

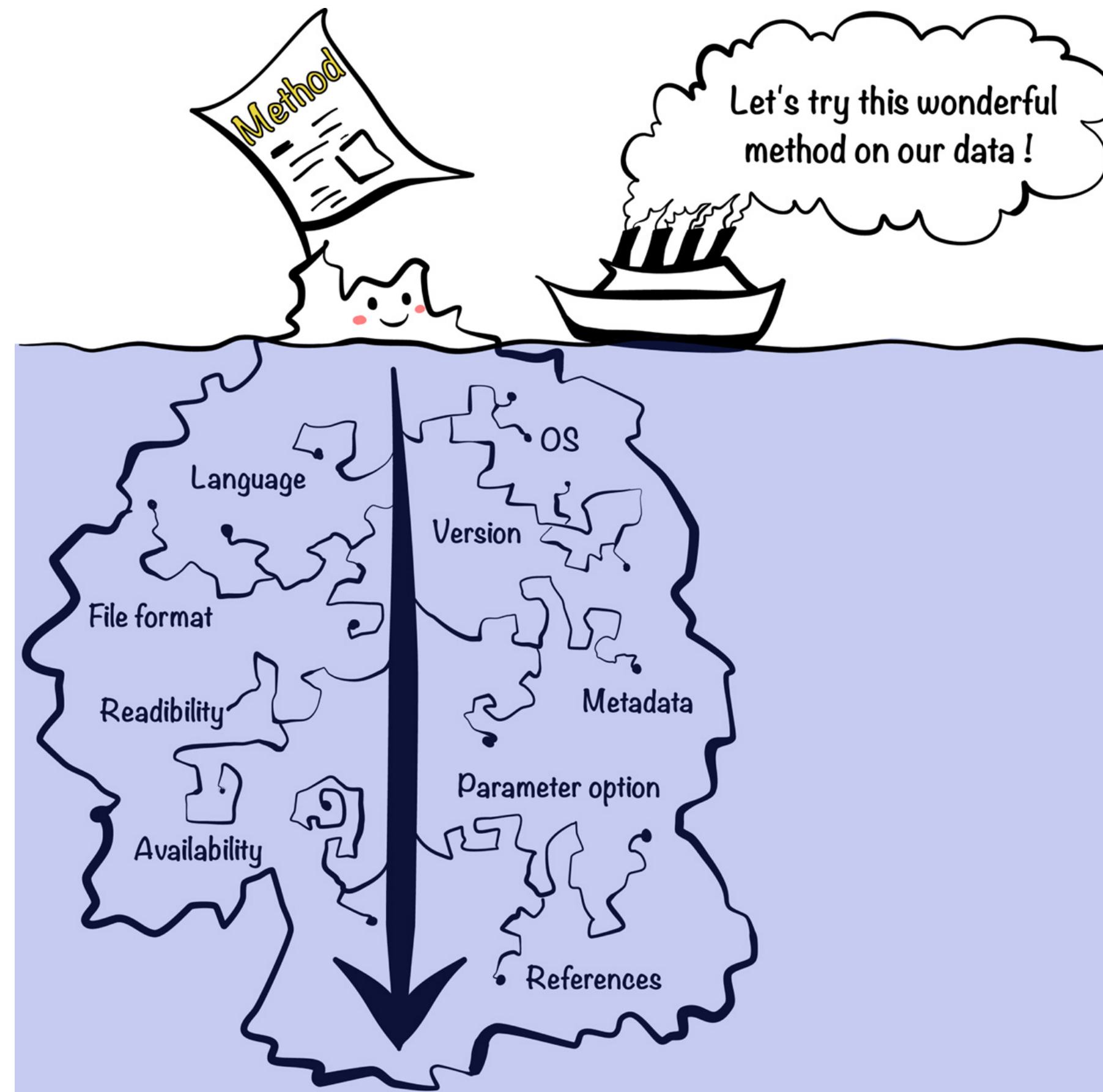


Others can replicate  
your findings with  
their data

# Reproducible bioinformatics

**Sharing your code is not enough**

# Reproducible bioinformatics

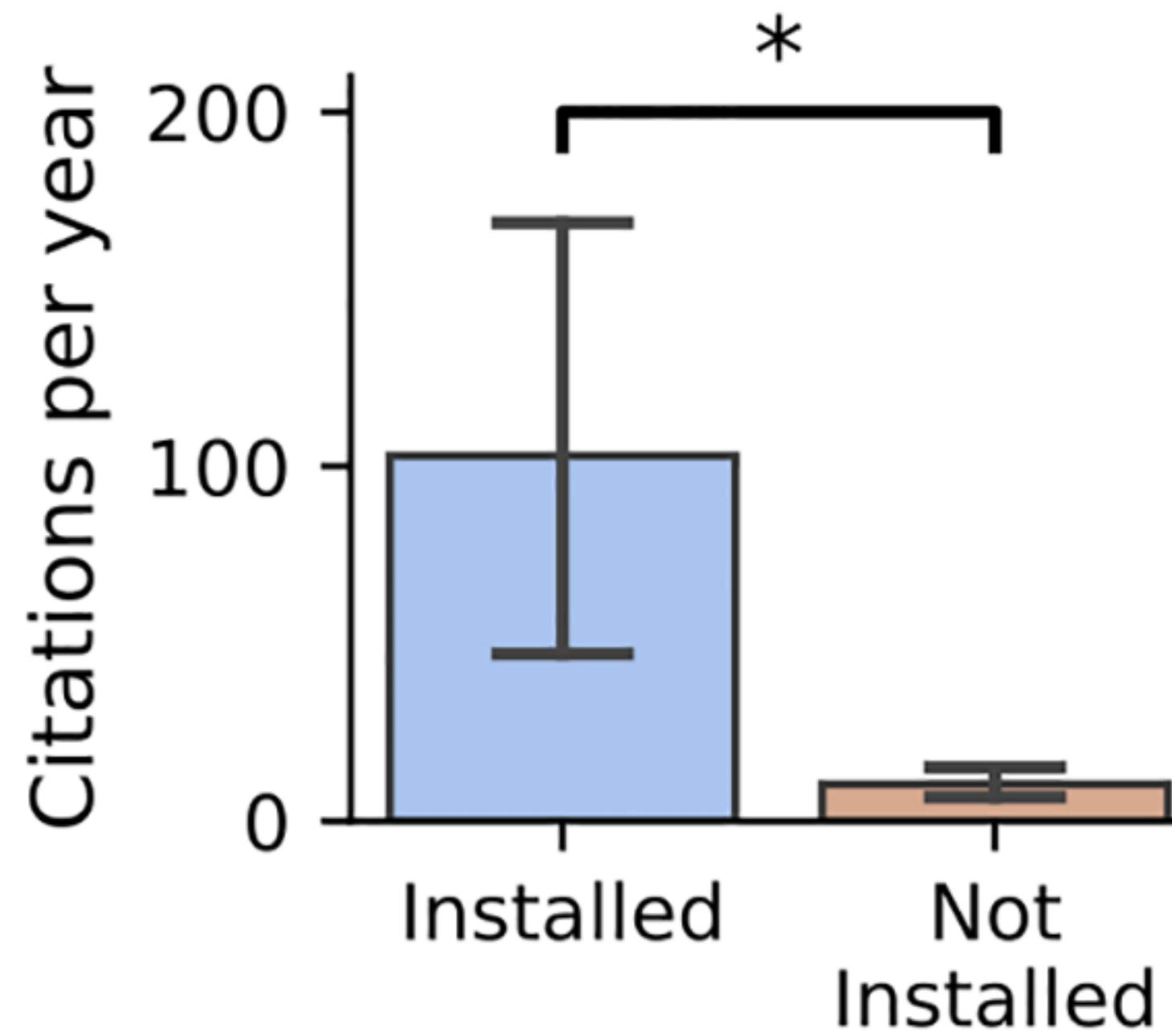


“  
First, we **tried** to re-run the analysis with the code and data provided by the authors.

Second, we **reimplemented** the whole method in a Python package...

”  
Experimenting with reproducibility:  
a case study of robustness in bioinformatics  
Kim et al., GigaScience (2018).  
<https://doi.org/10.1093/gigascience/giy077>

# Reproducible bioinformatics



“

We found that **28%** of all omics software resources are currently **not accessible** through URLs published in the paper.

Among the tools selected, **49%** were **difficult to install or could not be installed at all**.

”

Challenges and recommendations to improve the installability and archival stability of omics computational tools

Serghei Mangul, et al. PLOS Bio (2019).

<https://doi.org/10.1371/journal.pbio.3000333>

# Reproducible bioinformatics

Code

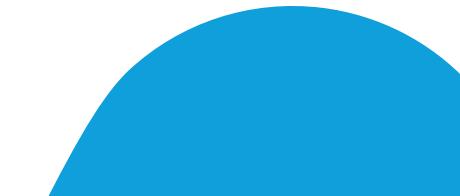
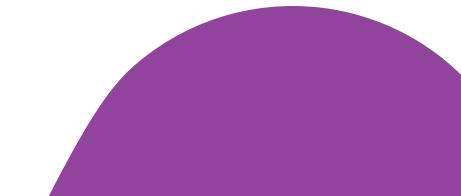
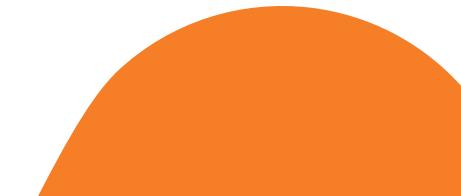
Custom scripts

Software

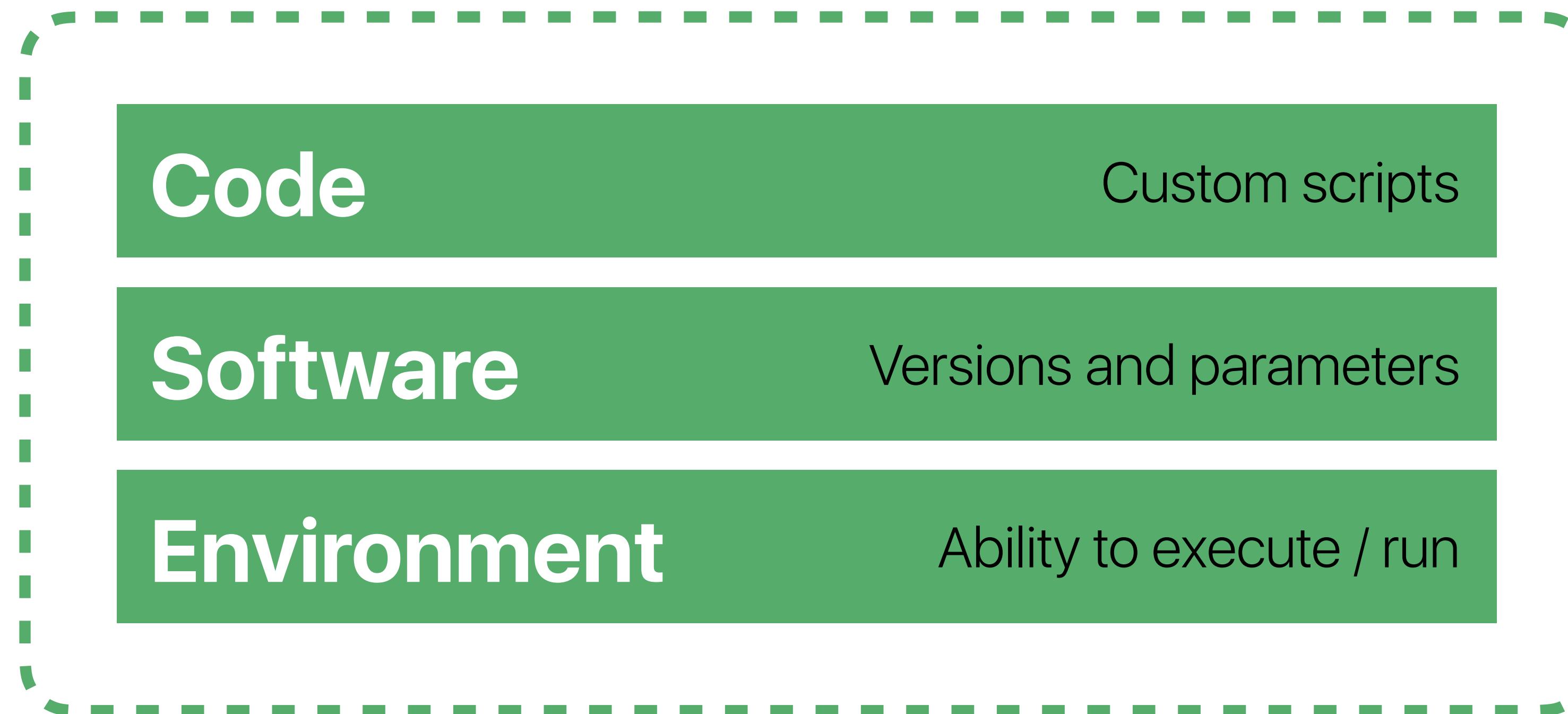
Versions and parameters

Environment

Ability to execute / run



# Reproducible bioinformatics



**Workflow**

nextflow





Language

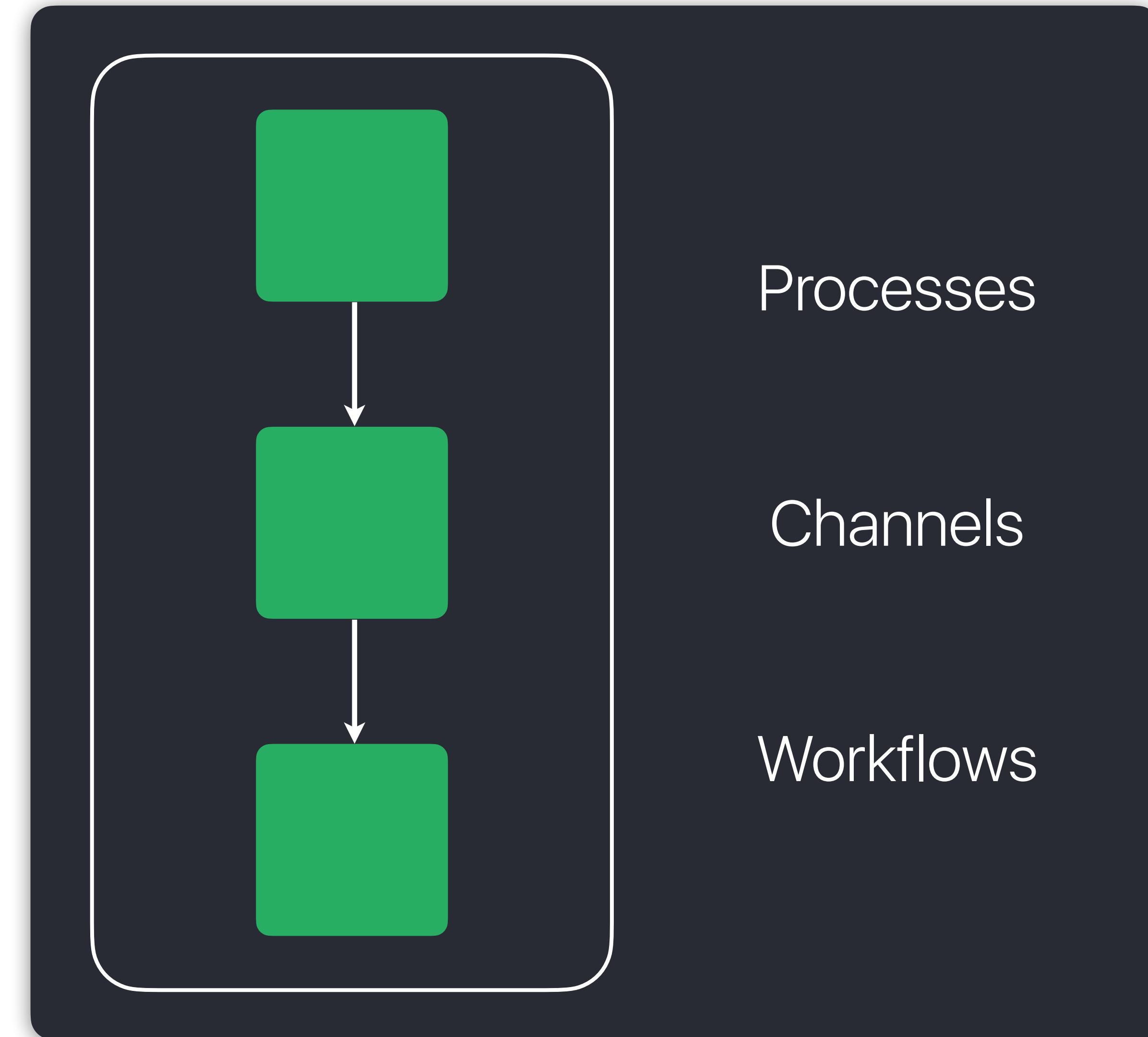
Software

Compute



Language

# nextflow



# nextflow

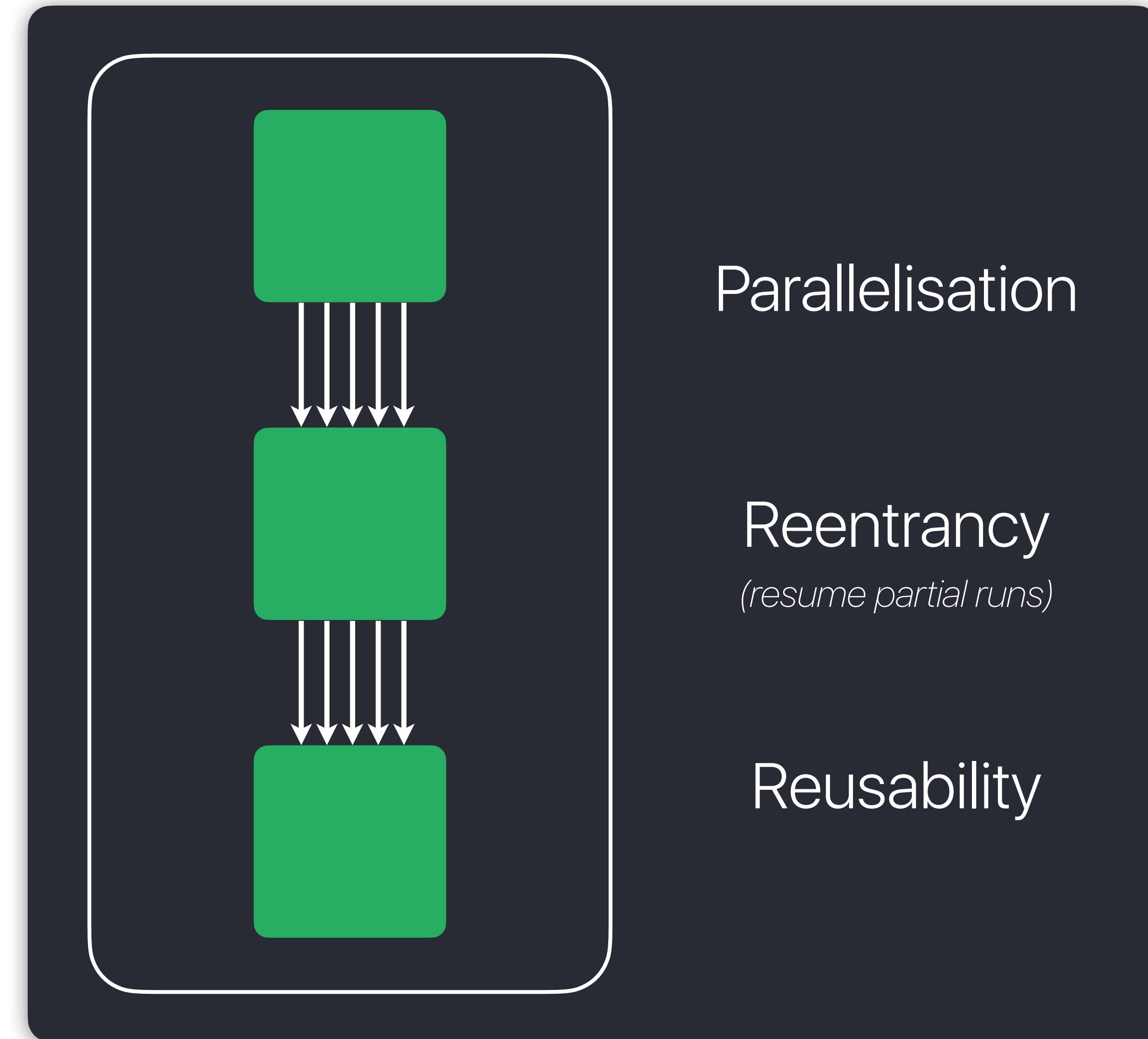
```
#!/usr/bin/env nextflow
process fastqc {
    input:
    path input

    output:
    path "*_fastqc.{zip,html}"

    script:
    """
    fastqc -q $input
    """
}

workflow {
    Channel.fromPath("*.fastq.gz") | fastqc
}
```

# nextflow





Language

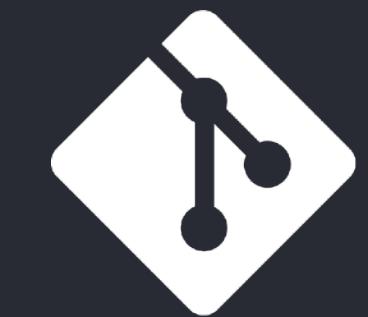


Language

Software

Compute

# nextflow



git



GitHub



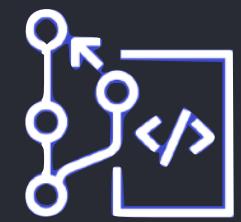
Bitbucket



GitLab



Gitea



AWS CodeCommit



Azure Repos

Software

Compute

# nextflow



git



GitHub

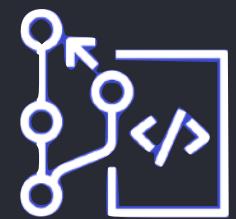
Bitbucket



GitLab



Gitea



AWS CodeCommit



Azure Repos



docker®



Singularity



Compute

# nextflow



git



GitHub



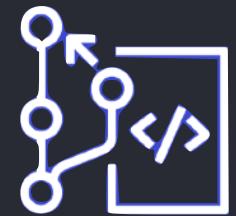
Bitbucket



GitLab



Gitea



AWS CodeCommit



Azure Repos



docker®



Singularity



SGE



Microsoft Azure



slurm

workload manager



aws

LSF

PBS



Google Cloud



kubernetes

# nextflow

```
#!/usr/bin/nextflow
process fastqc {
    container "biocontainers/fastqc" ←
        input:
        path input

        output:
        path "*_fastqc.{zip,html}"

        script:
        """
        fastqc -q $input
        """
}

workflow {
    Channel.fromPath("*.fastq.gz") | fastqc
}
```

```
docker {
    enabled = true
}
```



```
nextflow run main.nf -c ~/mylocal.conf
```

# nextflow

```
#!/usr/bin/nextflow
process fastqc {
    container "biocontainers/fastqc"

    input:
    path input

    output:
    path "*_fastqc.{zip,html}"

    script:
    """
    fastqc -q $input
    """
}

workflow {
    Channel.fromPath("*.fastq.gz") | fastqc
}
```

```
process {
    executor = 'slurm'
}
singularity {
    enabled = true
}
```



Singularity

```
nextflow run main.nf -c ~myhpc.conf
```

# nextflow

```
#!/usr/bin/nextflow
process fastqc {
    container "biocontainers/fastqc"
    input:
        path input
    output:
        path "*_fastqc.{zip,html}"
    script:
        """
        fastqc -q $input
        """
}
workflow {
    Channel.fromPath("*.fastq.gz") | fastqc
}
```

```
process {
    executor = 'slurm'
    queue = { task.time < 3.h ? 'short' : 'long' }
    beforeScript = "module load singularity"
}
singularity {
    enabled = true
    cacheDir = "/resources/nxf/singularity"
}
params {
    max_cpus = 24
    max_memory = 240.GB
    max_time = 168.h
}
```

```
nextflow run main.nf -c ~/myhpc.conf
```

# nextflow

```
#!/usr/bin/nextflow
process fastqc {
    container "biocontainers/fastqc"

    input:
    path input

    output:
    path "*_fastqc.{zip,html}"

    script:
    """
    fastqc -q $input
    """

}

workflow {
    Channel.fromPath("*.fastq.gz") | fastqc
}
```

```
process {
    executor = 'awsbatch'
    queue = 'my-batch-queue'
}
aws {
    region = 'us-east-1'
}
```



```
nextflow run main.nf -c ~/mycloud.conf
```

# nextflow

```
#!/usr/bin/nextflow
process fastqc {
    container "biocontainers/fastqc"

    input:
    path input

    output:
    path "*_fastqc.{zip,html}"

    script:
    """
        fastqc -q $input
    """

}

workflow {
    Channel.fromPath(params.input) | fastqc
}
```

```
params {
    input = "*.fastq.gz"
}
```

```
nextflow run main.nf --input "data/input*.fq"
```



# Reproducible

Between runs

# Portable

Between systems

**nextflow**

**nf-core**





A community effort to collect a curated set of analysis pipelines built using Nextflow.

<https://nf-co.re>

# nf-core

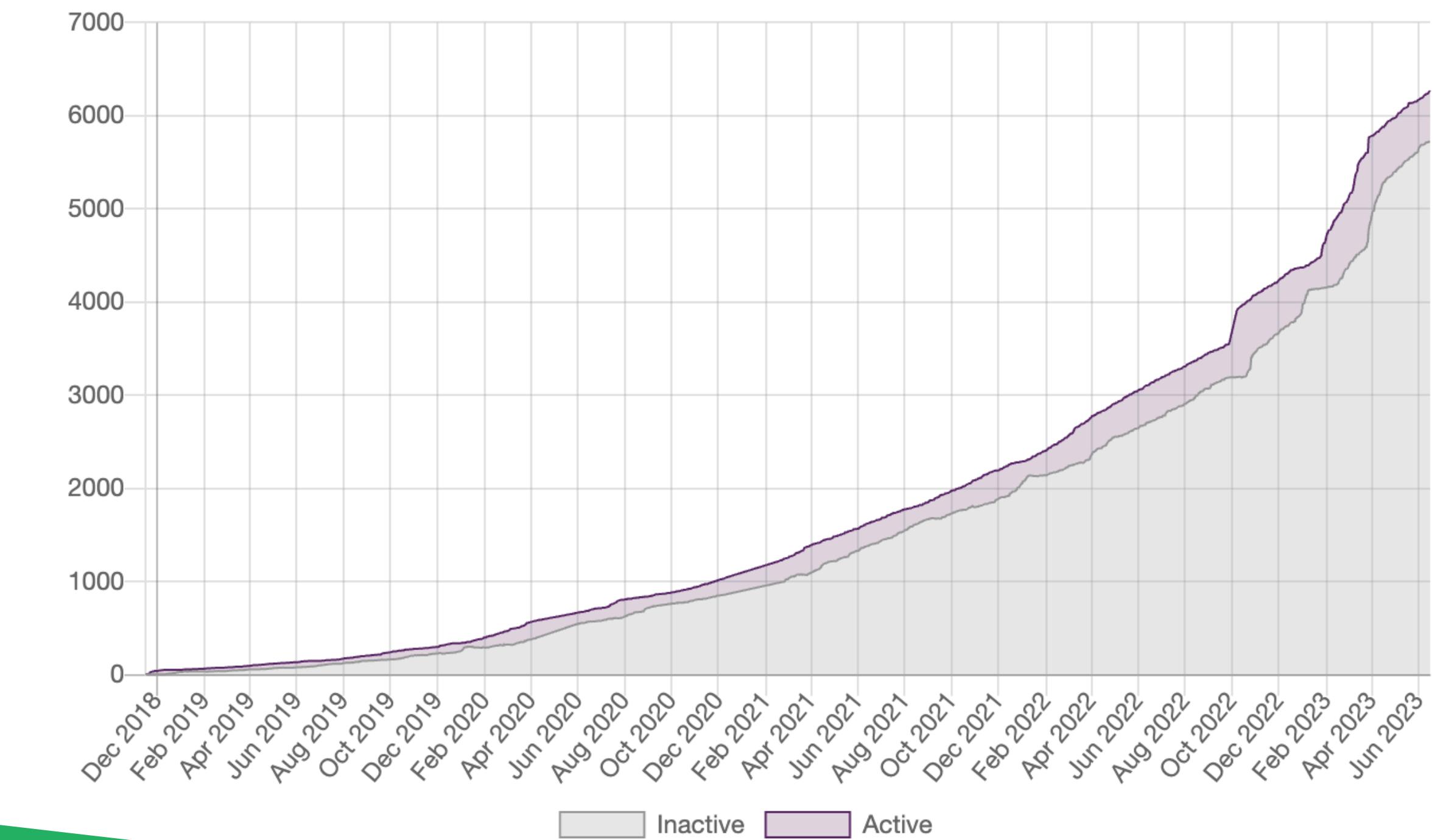


> 6000  
> 2000

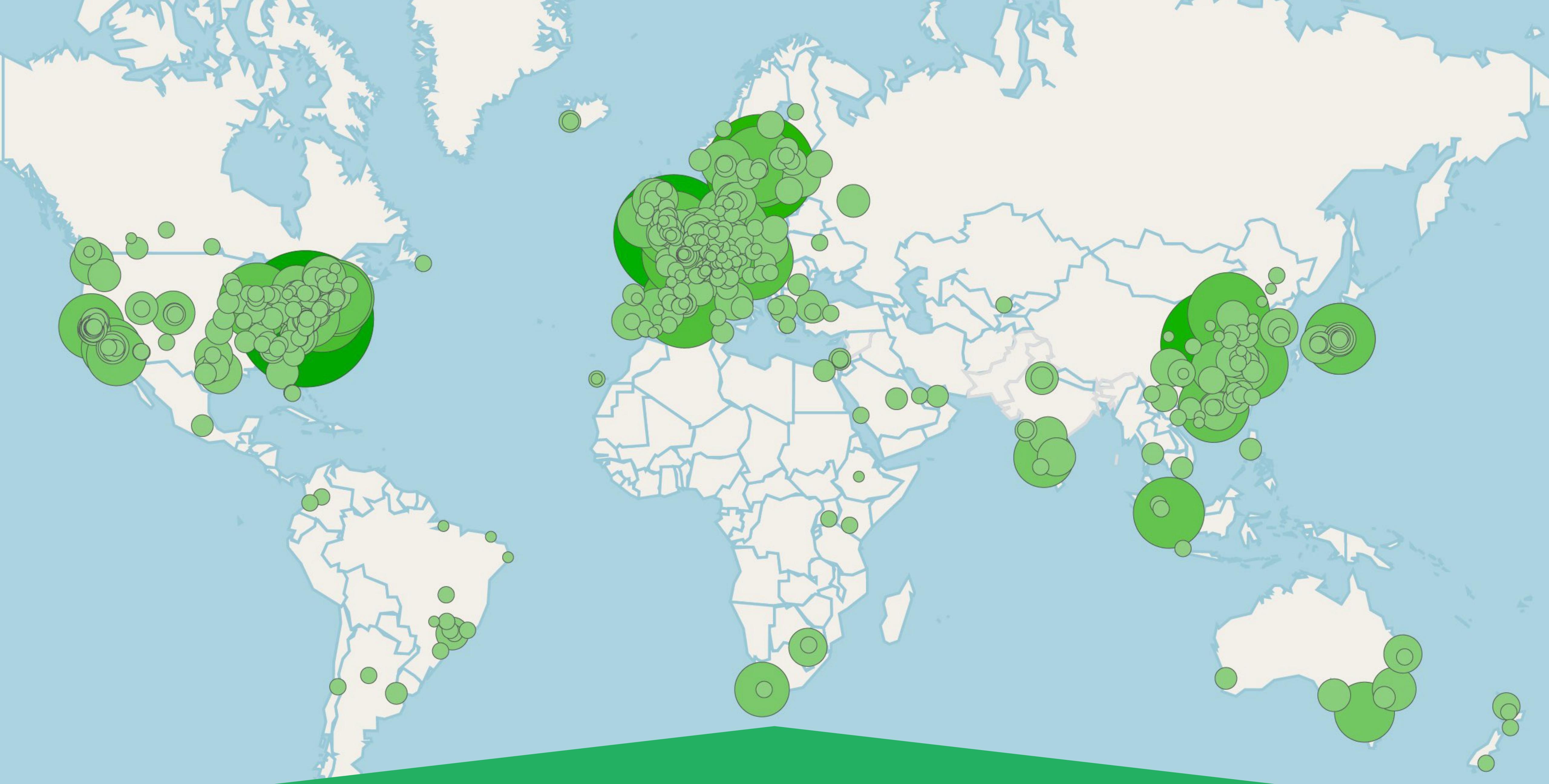
Slack members

GitHub contributors

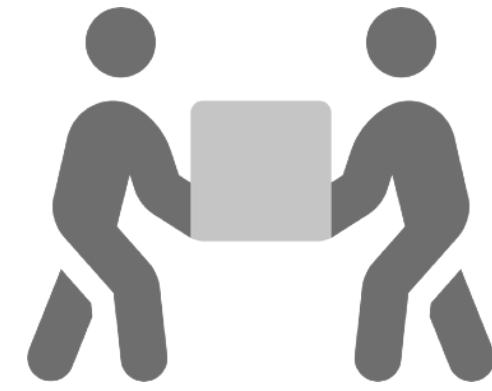
nf-core Slack users over time



<https://nf-co.re>



<https://nf-co.re>



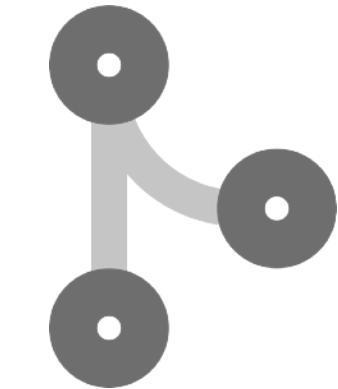
## Cooperation

Develop with the  
community



## Standards

Use a common  
template



## Collaboration

Collaborate, don't  
duplicate

<https://nf-co.re>

# nf-core



## Framework

Tools built for  
everyone



## Compatibility

Works with any Nextflow  
pipeline



## Components

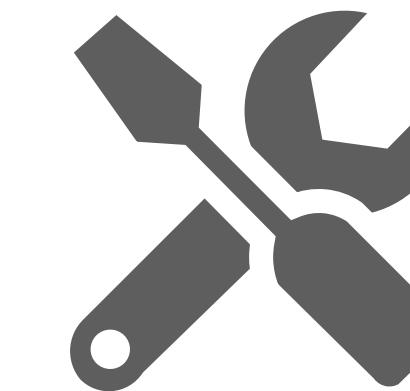
Collaborate on  
components

<https://nf-co.re>



## 82 Pipelines

Ready to use, covering most techniques in NGS

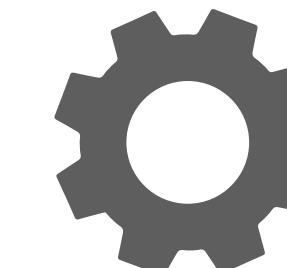


## Helper tools

Running pipelines, writing pipelines, testing & automation

## 957 Modules

Tool process wrappers, with software and CI testing



## 44 Subworkflows

Shared subworkflows for common analysis pathways

<https://nf-co.re>



**metaboigniter** Pre-processing of mass spectrometry-based metabolomics data

**funcscan** Mining for antimicrobial peptides, antibiotic resistance genes and biosynthetic gene clusters.

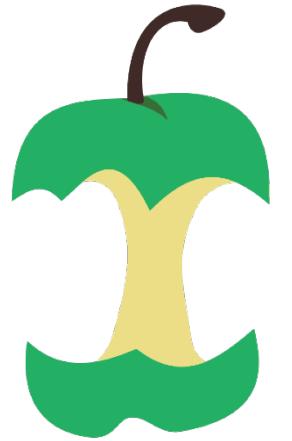
**methylseq** DNA methylation sequencing data (eg. Bisulfite)

**viralrecon** Assembly and intrahost/low-frequency variant calling for viral samples

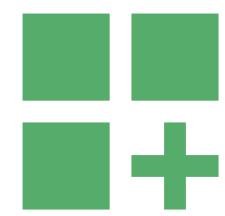
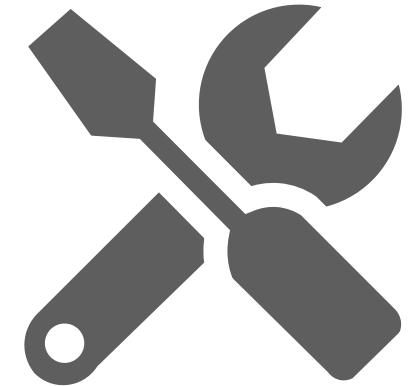
**rnaseq** RNA sequencing analysis pipeline with gene/isoform counts and extensive quality control.

**scrnaseq** A single-cell RNAseq pipeline for 10X genomics data

# nf-core

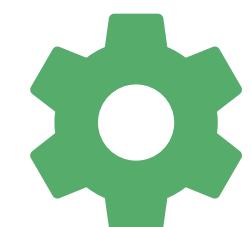


# Helper tools



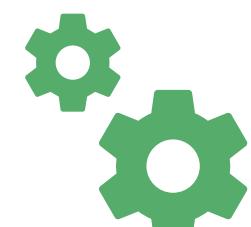
## Pipelines

Create from template,  
sync to get updates.



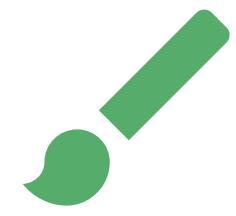
## Subworkflows

Create, install and  
update.



## Modules

Create, install, update,  
patch, test.



## Schema

Build your pipeline  
schema with a GUI.



## Linting

Test nf-core standards  
and best practices.

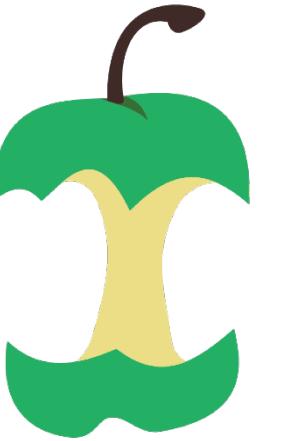


## Download

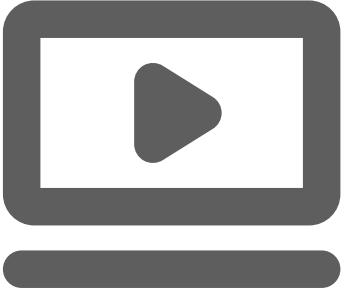
Fetch with singularity  
images for offline use.

<https://nf-co.re>

# nf-core



# Webinar



WEBINAR

nextflow

## Nextflow best practices: Leveraging nf-core tooling and standards within your organization

June 21 | 12:00 PM ET · 9:00 AM PT · 6:00 PM CEST

Virtual



Harshil Patel  
Seqera Labs



Phil Ewels  
Seqera Labs

[https://seqera.io/  
webinar-nextflow-nf-core/](https://seqera.io/webinar-nextflow-nf-core/)

<https://nf-co.re>

Correspondence | Published: 13 February 2020

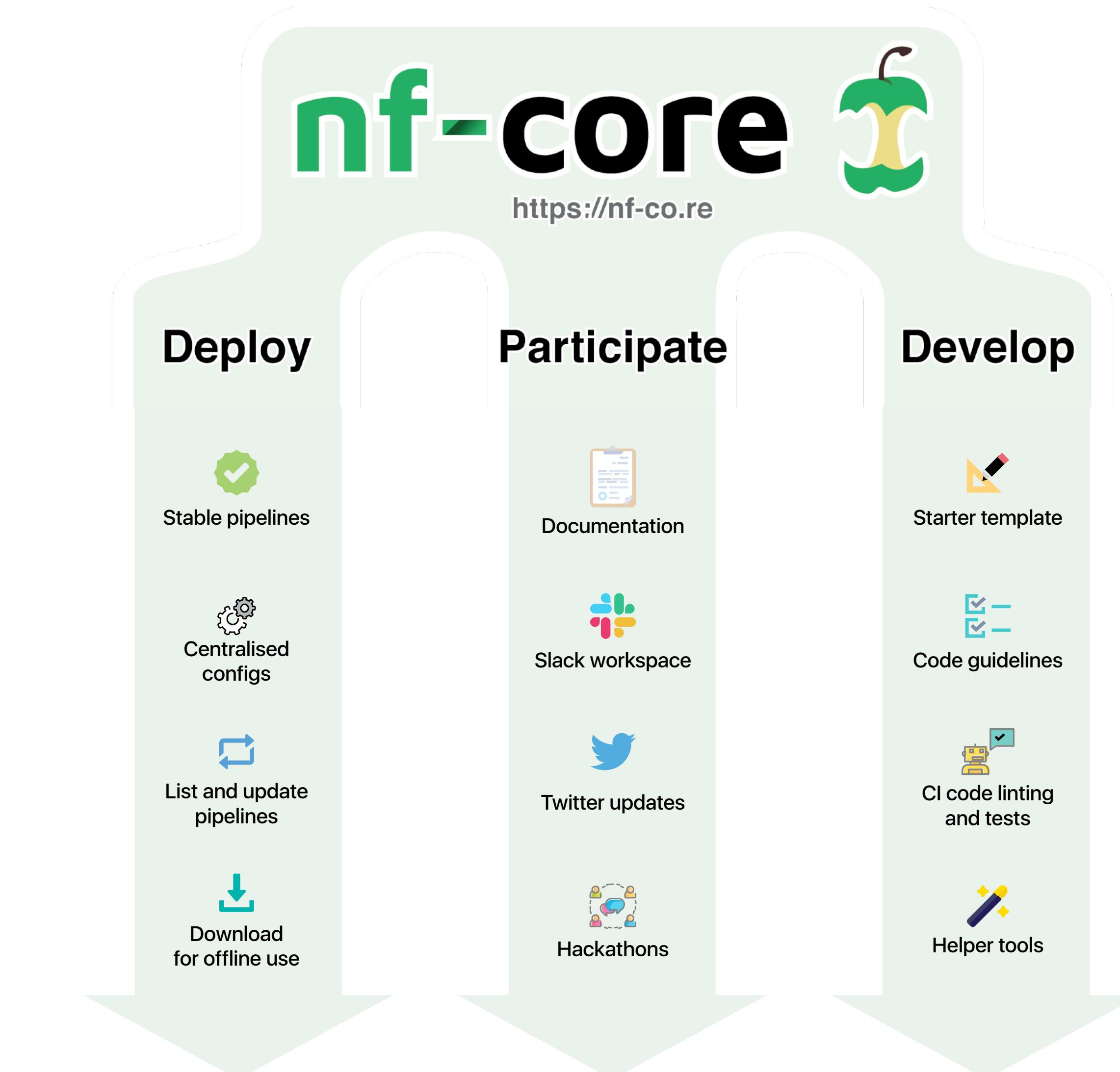
## The nf-core framework for community-curated bioinformatics pipelines

Philip A. Ewels, Alexander Peltzer, Sven Fillinger, Harshil Patel, Johannes Alneberg,  
Andreas Wilm, Maxime Ulysse Garcia, Paolo Di Tommaso & Sven Nahnsen 

*Nature Biotechnology* 38, 276–278(2020) | Cite this article

3253 Accesses | 3 Citations | 172 Altmetric | Metrics

**To the Editor** — The standardization, portability and reproducibility of analysis pipelines are key issues within the bioinformatics community. Most bioinformatics pipelines are designed for use on-premises; as a result, the associated software dependencies and execution logic are likely to be tightly coupled with proprietary computing environments. This can make it difficult or even impossible for others to reproduce the ensuing results, which is a fundamental requirement for the validation of scientific findings. Here, we introduce the nf-core framework as a means for the development of collaborative, peer-reviewed, best-practice analysis pipelines (Fig. 1). All nf-core pipelines are written in Nextflow and so inherit the ability to be executed on most computational infrastructures, as well as having native support for container technologies such as Docker and Singularity. The nf-core community (Supplementary Fig. 1) has developed a suite of tools that automate pipeline creation, testing, deployment and synchronization. Our goal is to provide a framework for high-quality bioinformatics pipelines that can be used across all institutions and research facilities.

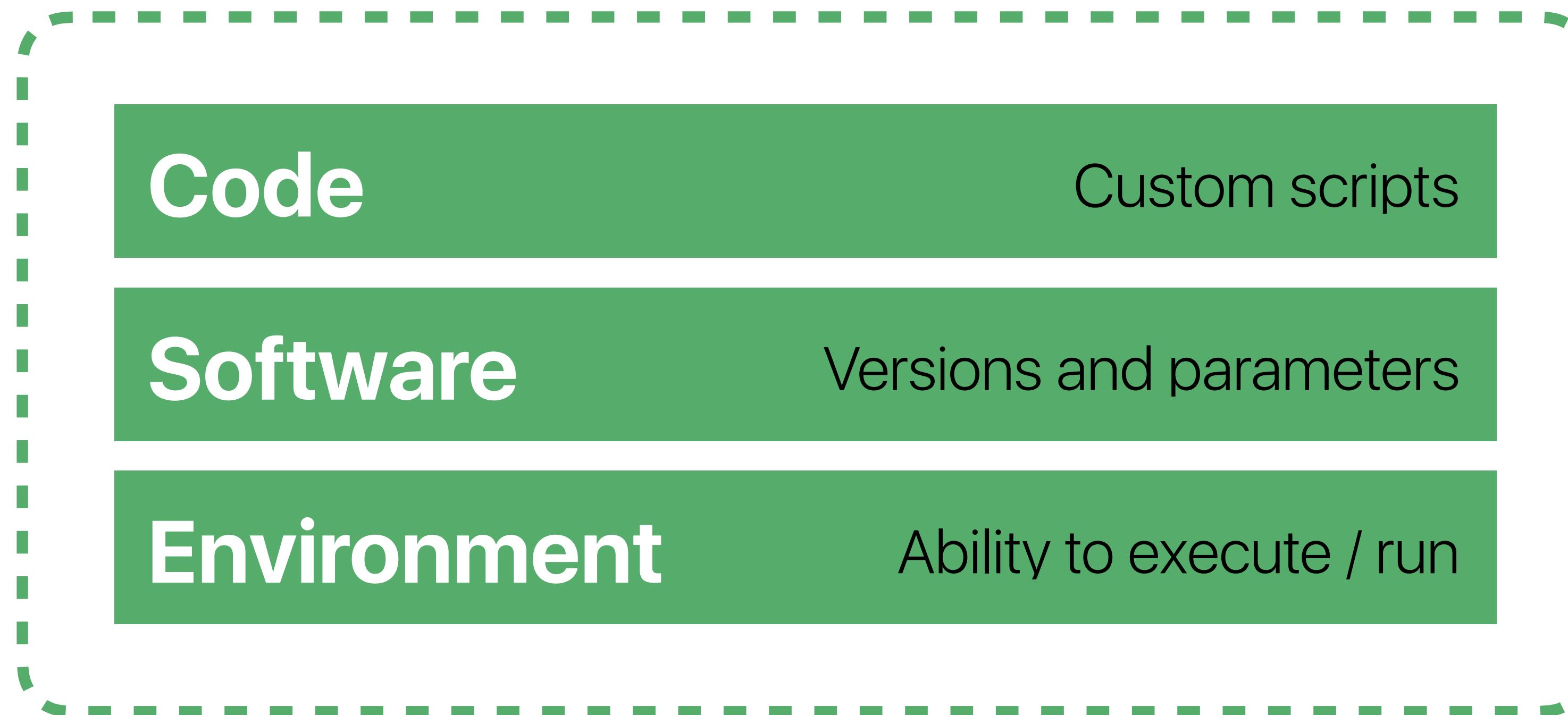


# Join the community

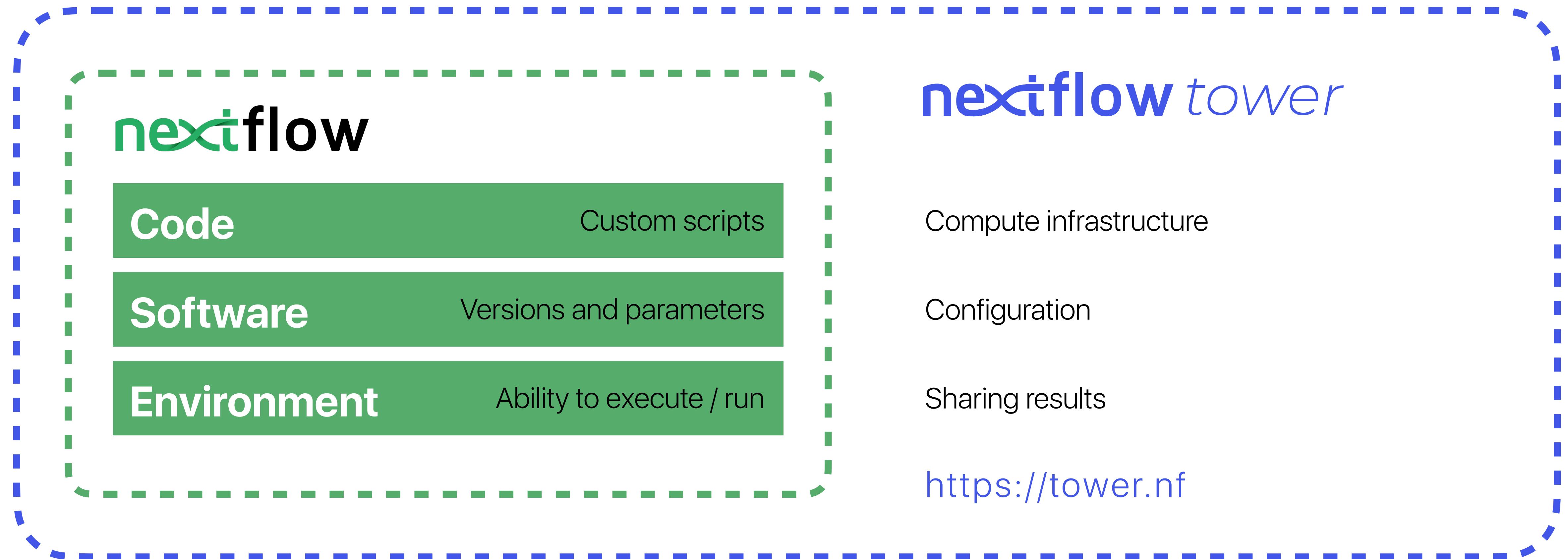


<https://nf-co.re/join>

# Reproducible bioinformatics



# Reproducible bioinformatics for everyone



# nextflow tower

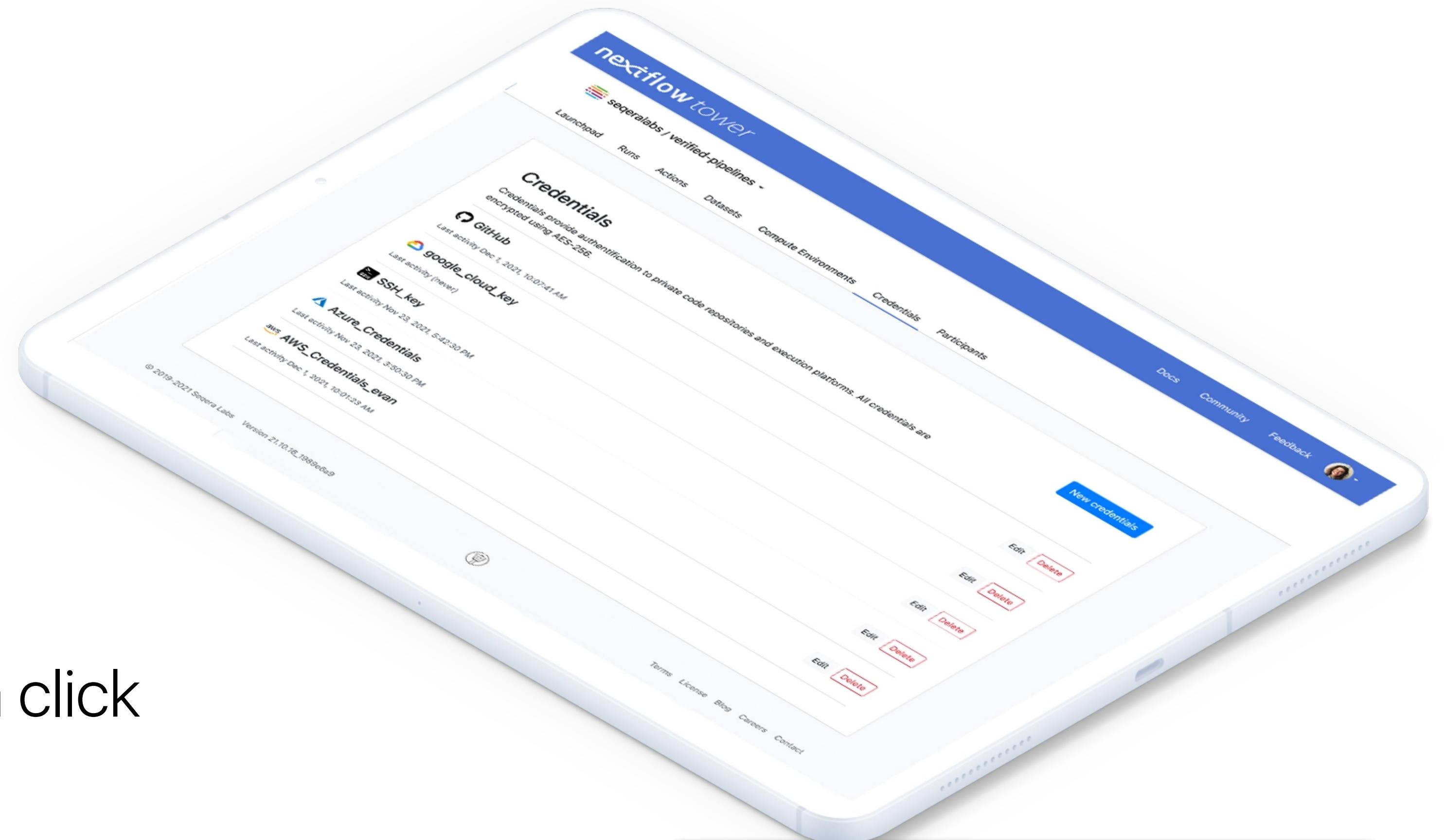


Intuitive launchpad interface

Launch, manage, and monitor

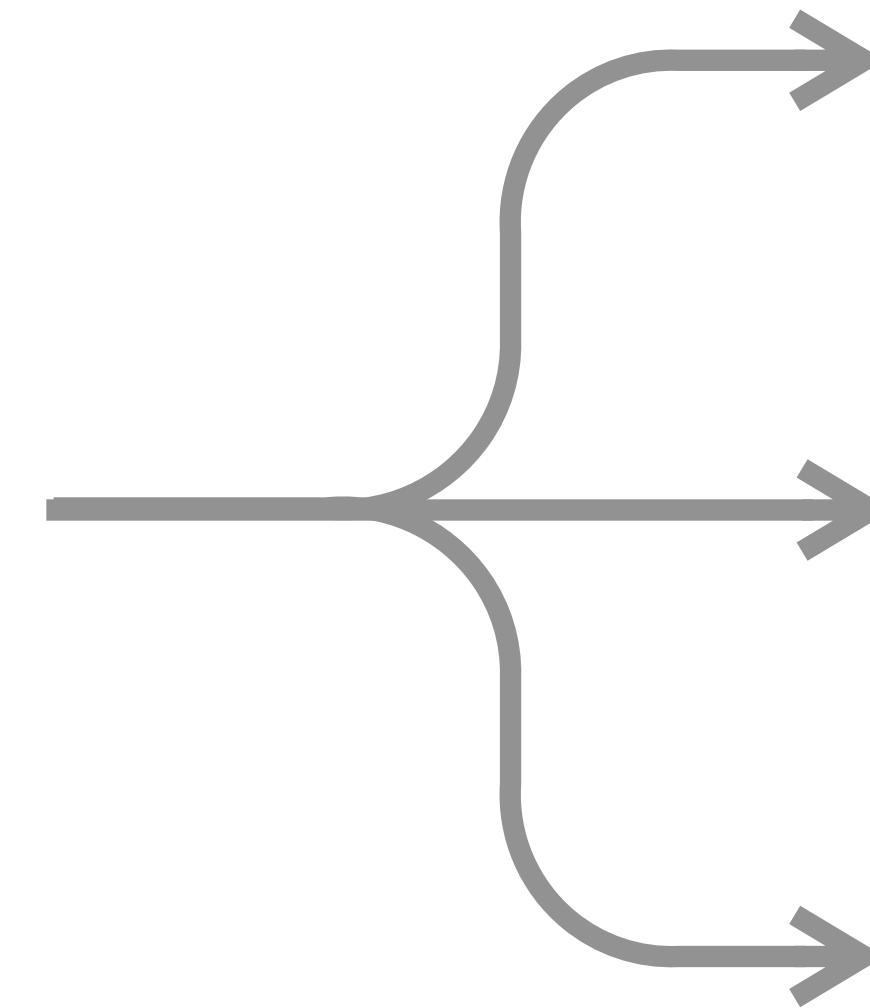
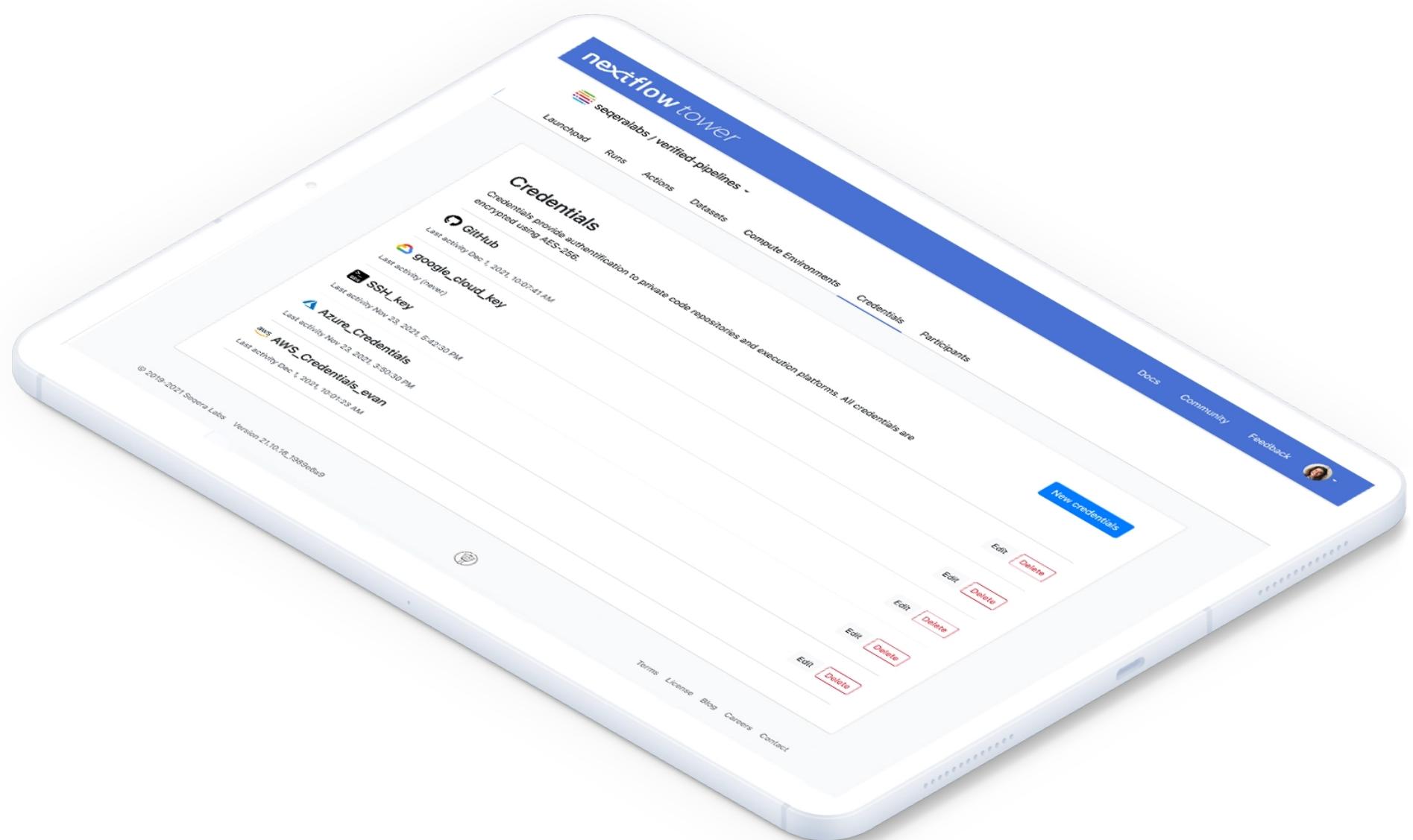
Share runs and work in teams

Create cloud infrastructure with a click



<https://tower.nf>

# nextflow tower



**Community:** Open source

**Cloud:** Free & paid tiers

**Enterprise:** Commercial

<https://tower.nf>

# Seqera Platform

Streamlining the scientific data analysis life cycle



## Data Developer

Tools & workflows  
IDEs - VSCode  
CI/CD  
Containers

## Data Explorer

Instruments  
Cloud storage  
File systems  
LIMS  
Public resources

### Nextflow

Scientific workflow management

### MultiQC

Bioinformatics reporting

### Fusion

File-system for cloud-native data pipelines

### Wave

Container provisioning

## Data Analytics

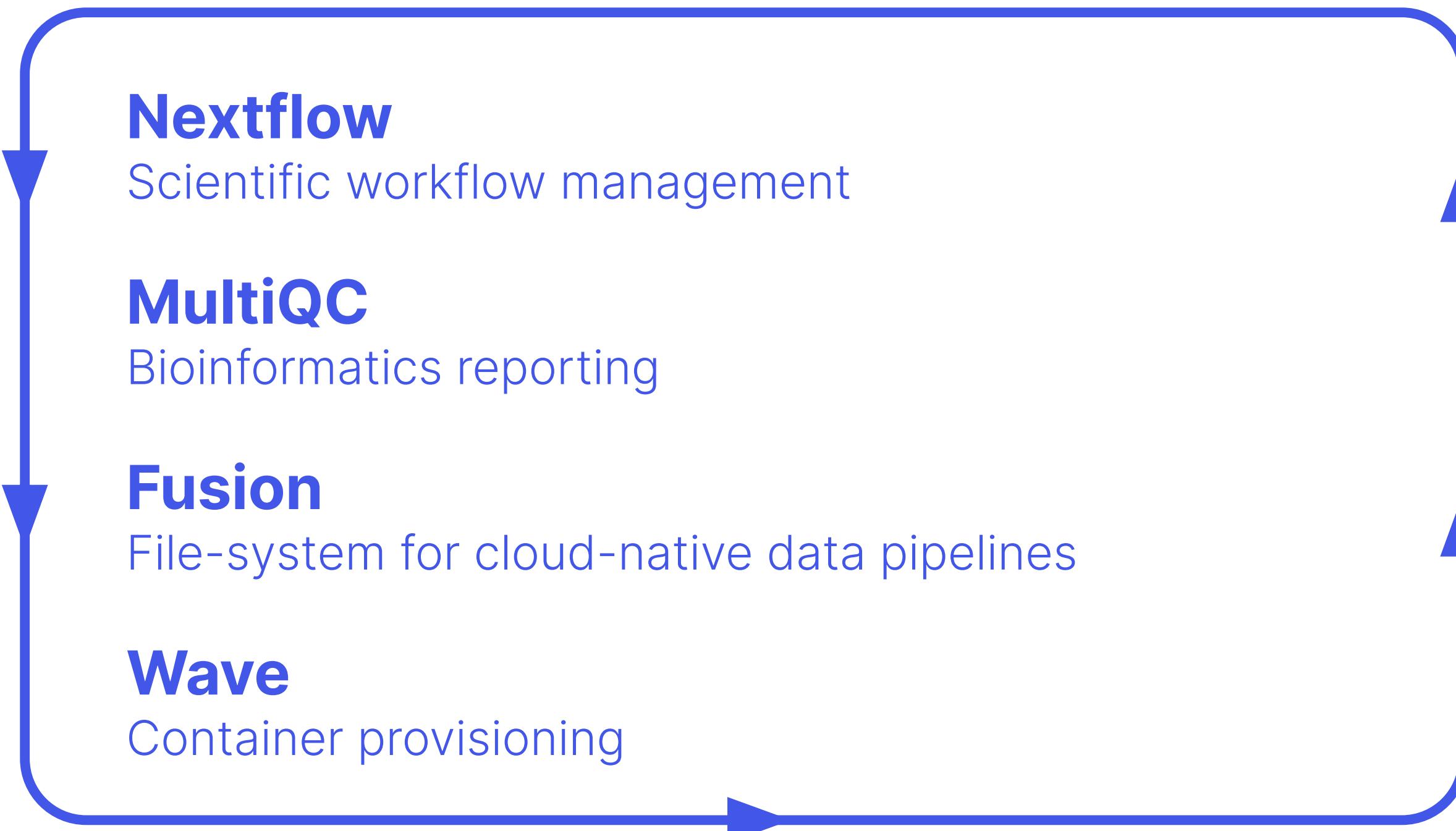
Reporting  
Metrics  
Provenance  
Compliance

## Data Studio

Interactive Notebooks  
RStudio  
RShiny Applications  
Genome Browsers

## Data Pipelines

Batch Compute  
Orchestration  
Performant Storage



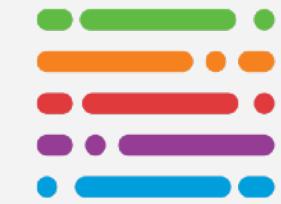
# Phil Ewels

<https://phil.ewels.co.uk>

[phil@seqera.io](mailto:phil@seqera.io)

 tallphil

 ewels



**seqeralabs**

<https://seqera.io>

**Chan Zuckerberg  
Initiative**



## Basic training

Free to watch online

[youtube.com/@nf-core](https://youtube.com/@nf-core)

**nextflow**  
**SUMMIT 2023**

<https://summit.nextflow.io>

## Advanced training

27-28 September 2023

**Barcelona Summit 2023**

16-20 October 2023

<https://nf-co.re>

<https://nextflow.io>