Desenho de *pipelines* para análise reprodutível e controlada de dados de sequenciação

Hugo Martiniano

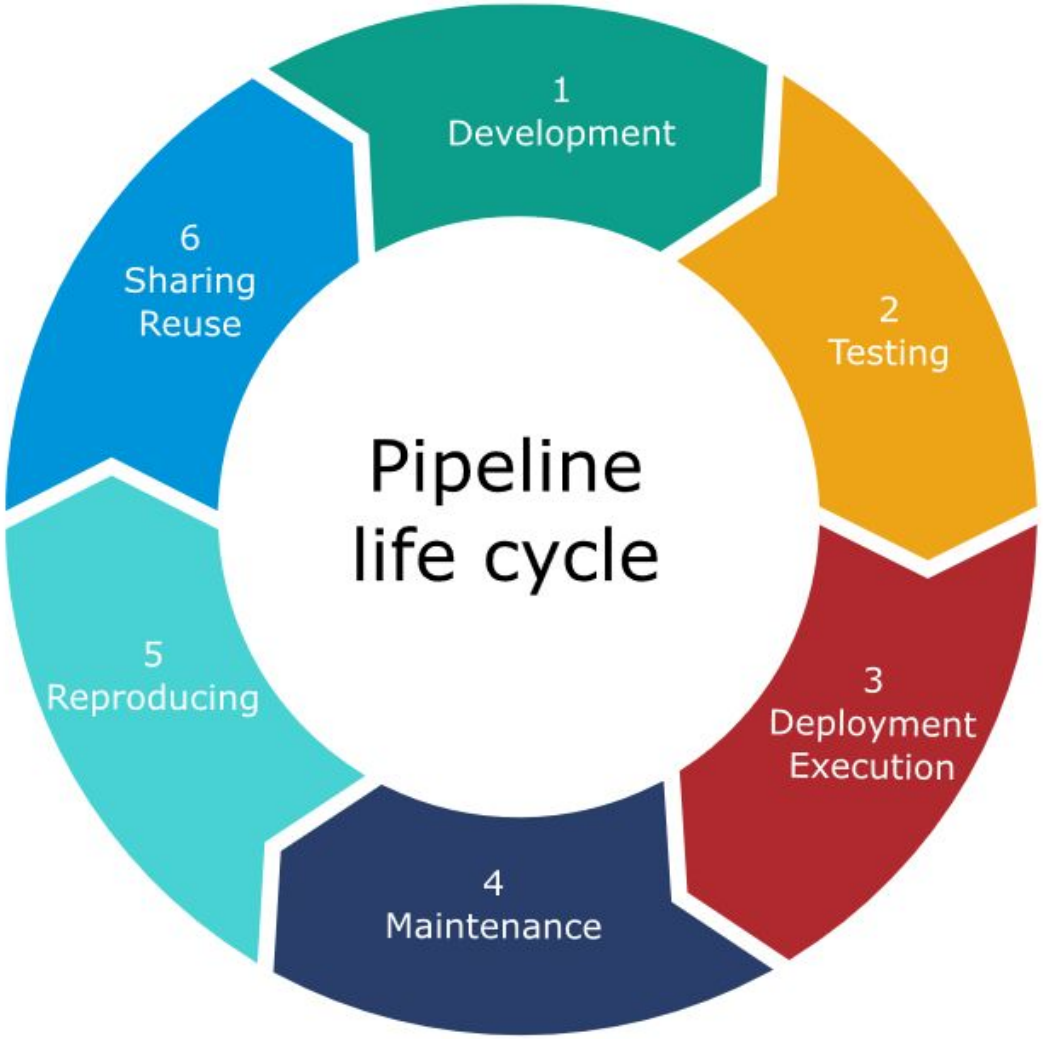
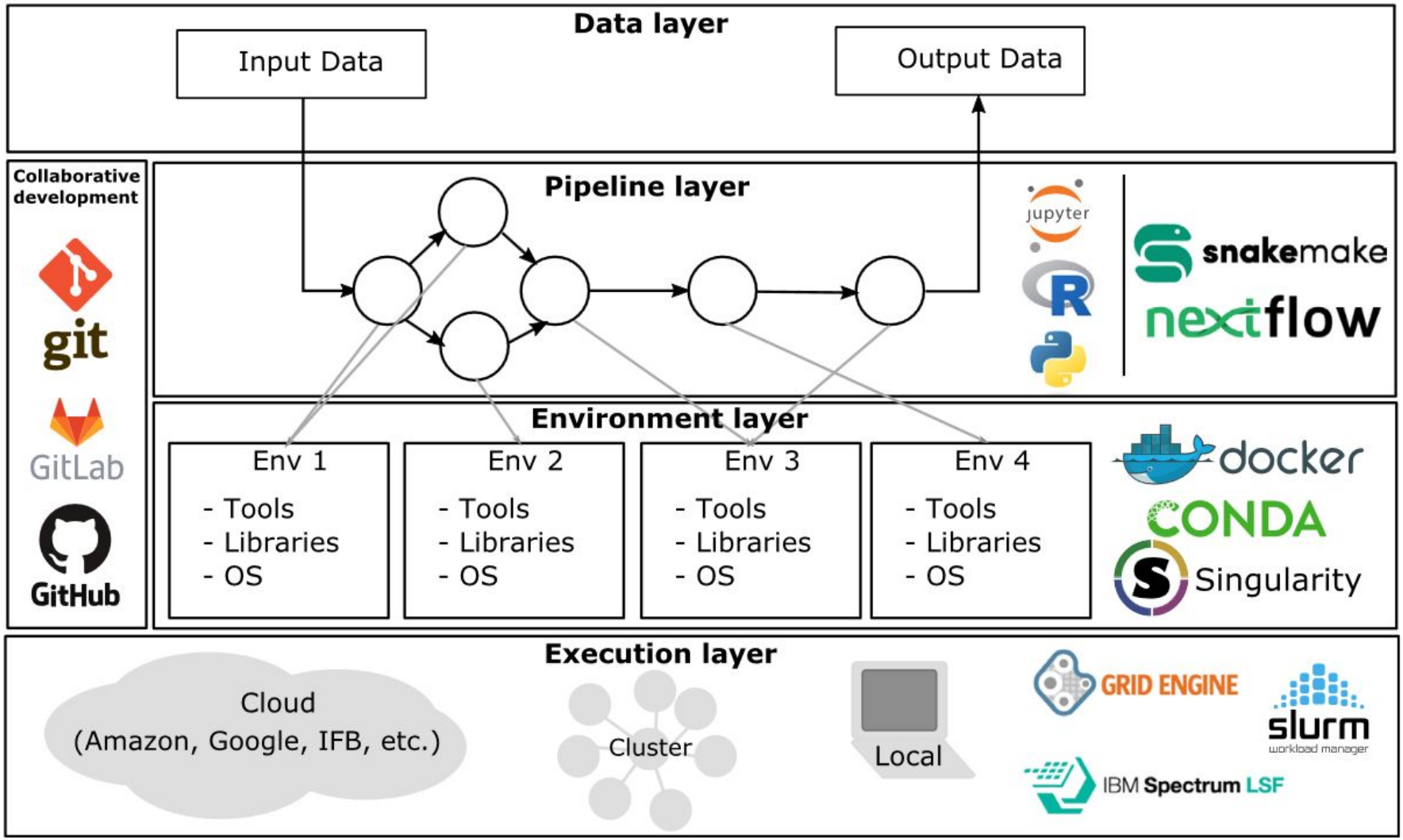


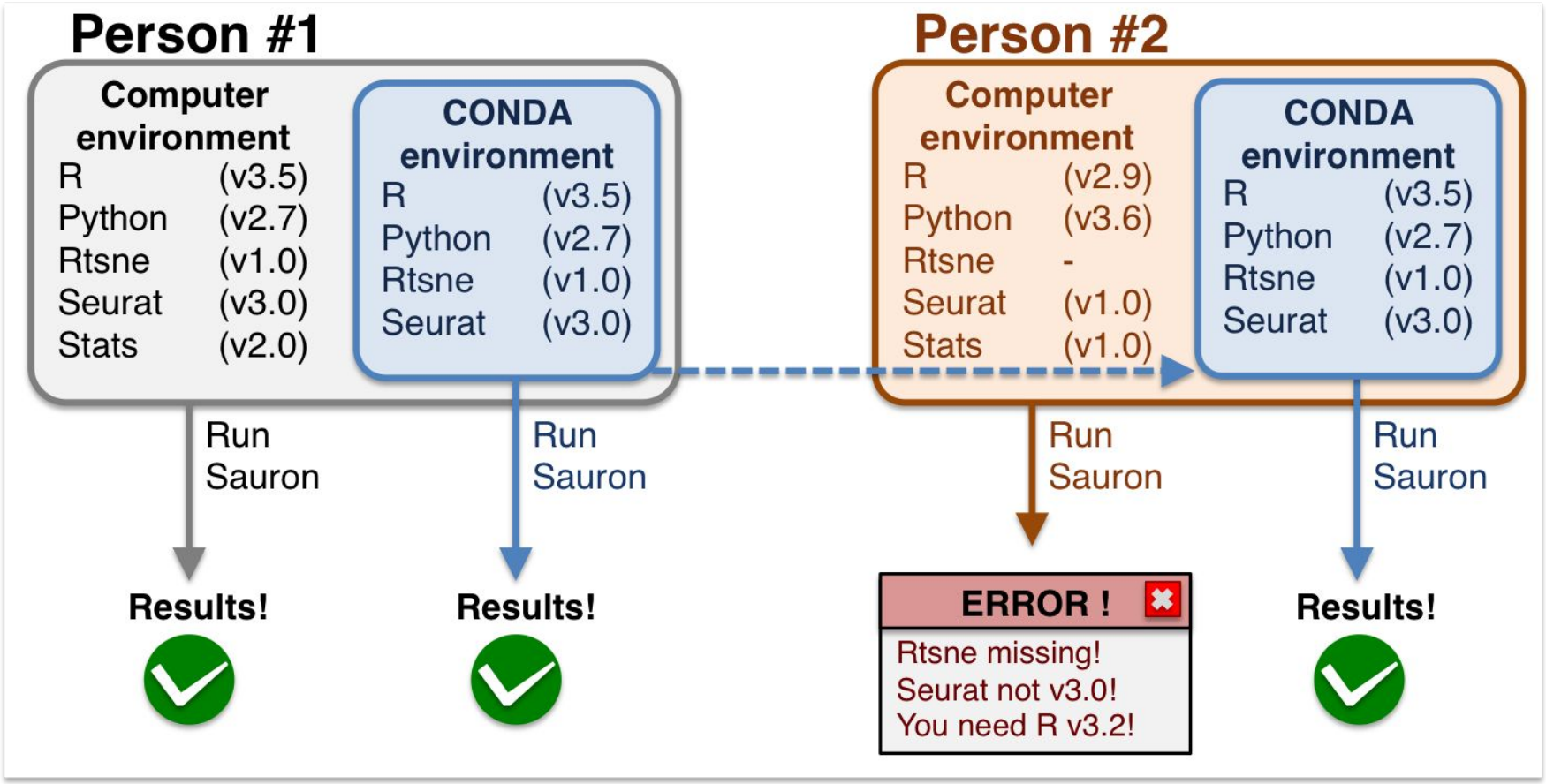
Fig. 1. Life cycle of a bioinformatics pipeline.

https://doi.org/10.1016/j.csbj.2023.03.003

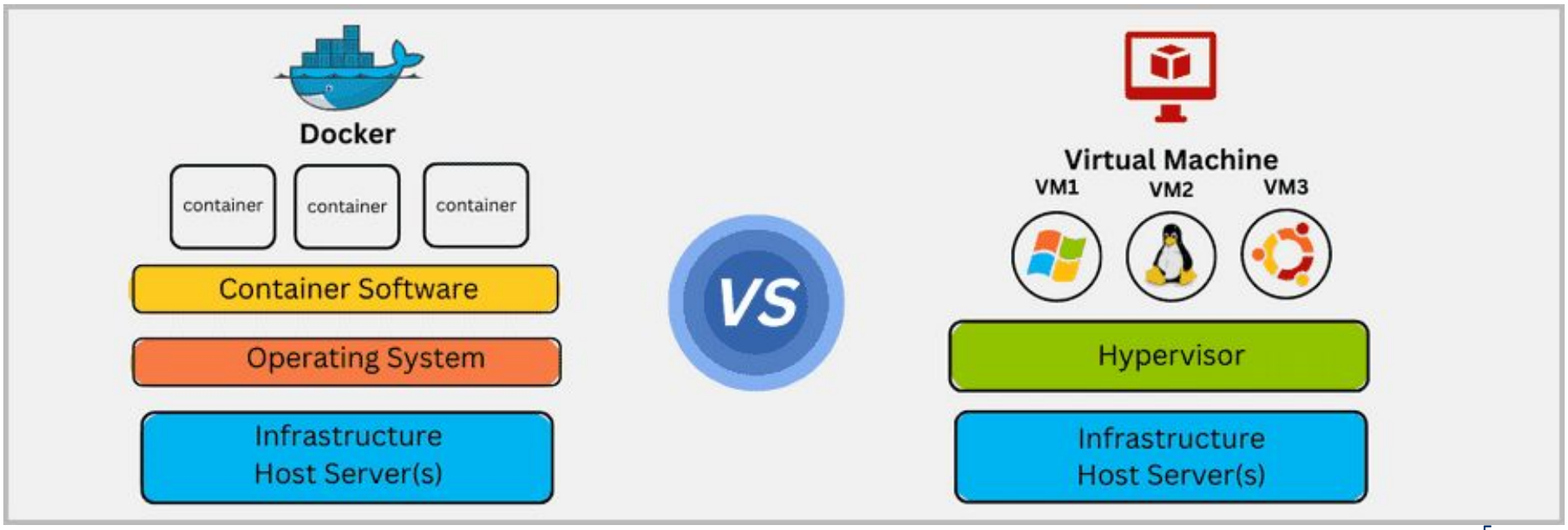


https://doi.org/10.1016/j.csbj.2023.03.003

CONDA


Person #1
Computer environment
R (v3.5)
Python (v2.7)
Rtsne (v1.0)
Seurat (v3.0)
Stats (v2.0)
CONDA environment
R (v3.5)
Python (v2.7)
Rtsne (v1.0)
Seurat (v3.0)
Run Sauron
Run Sauron
Results!
Results!
Person #2
Computer environment
R (v2.9)
Python (v3.6)
Rtsne -
Seurat (v1.0)
Stats (v1.0)
CONDA environment
R (v3.5)
Python (v2.7)
Rtsne (v1.0)
Seurat (v3.0)
Run Sauron
Run Sauron
ERROR !
Rtsne missing!
Seurat not v3.0!
You need R v3.2!
Results!

Contentores - Docker/Singularity



Gestores de workflow

- Existem mais de 100 gestores de workflow diferentes (https://github.com/pditommaso/awesome-pipeline)

- Várias áreas de aplicação têm necessidades diferentes

- Principais vantagens:
  - Reprodutibilidade
  - Execução automática
  - Mais eficiência (paralelização ou optimização)
  - Escalabilidade e Interoperabilidade

Gestores de workflow

nextflow

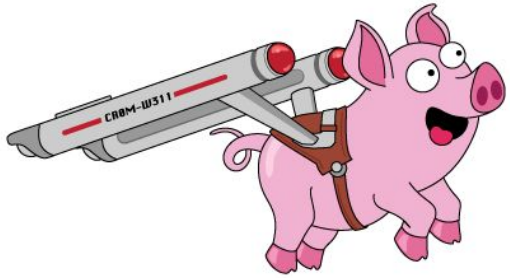
https://www.nextflow.io/

snakemake
A framework for reproducible data analysis

https://snakemake.github.io/

Galaxy

https://galaxyproject.org/

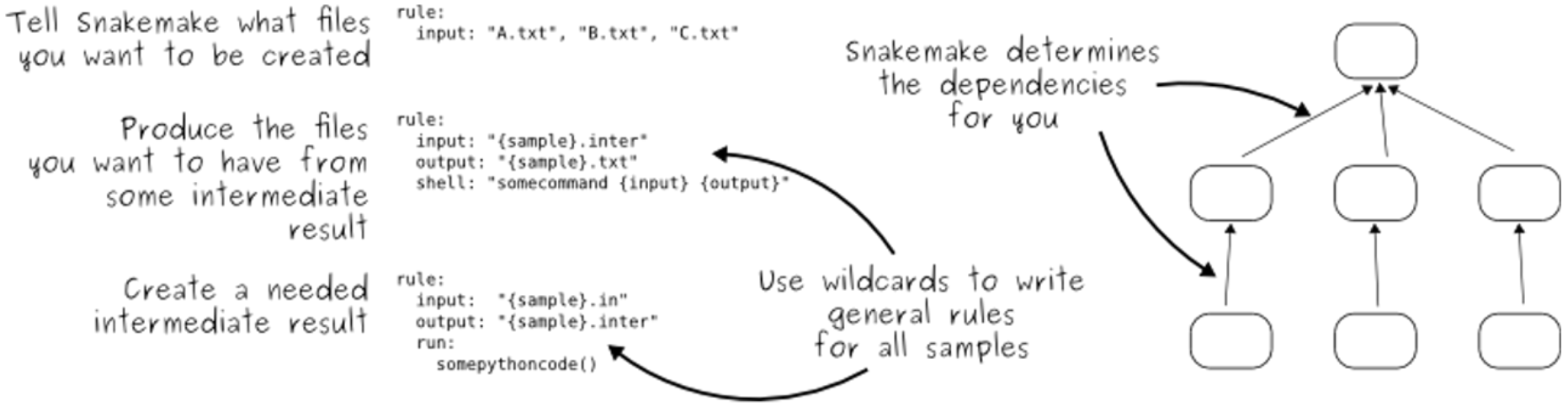


Cromwell

http://cromwell.readthedocs.io/

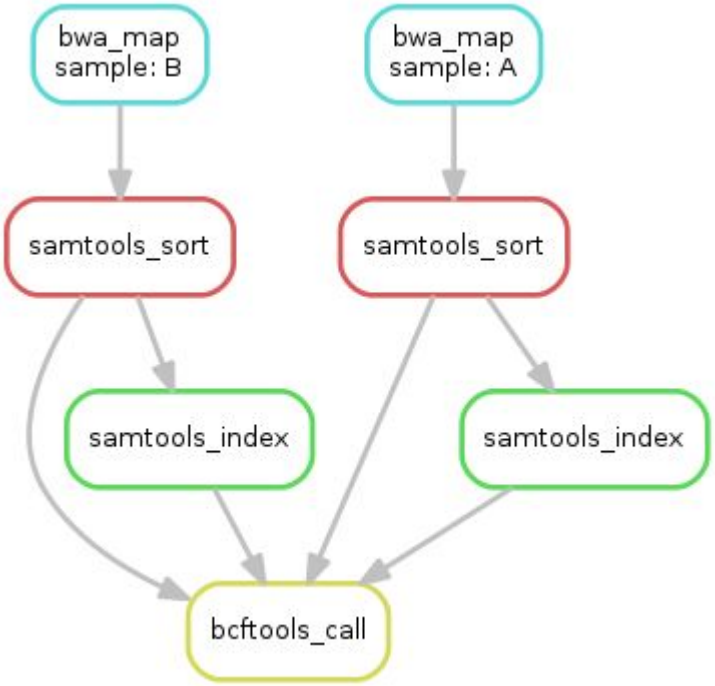
Snakemake

- **Linguagem Declarativa**
- **Reprodutibilidade - Integração com o conda, docker ou singularity**
- **Paralelização automática**
- **Controle de Recursos (CPU, memória, tempo de execução, etc)**
- **Comunidade ativa e Documentação**
- **Portabilidade e escalabilidade**



Snakemake - Exemplo

- **Pipeline com 3 passos**
  - **alinhamento**
  - **indexação**
  - **chamada de variantes**



https://snakemake.readthedocs.io/en/stable/tutorial/basics.html#step-1-mapping-reads

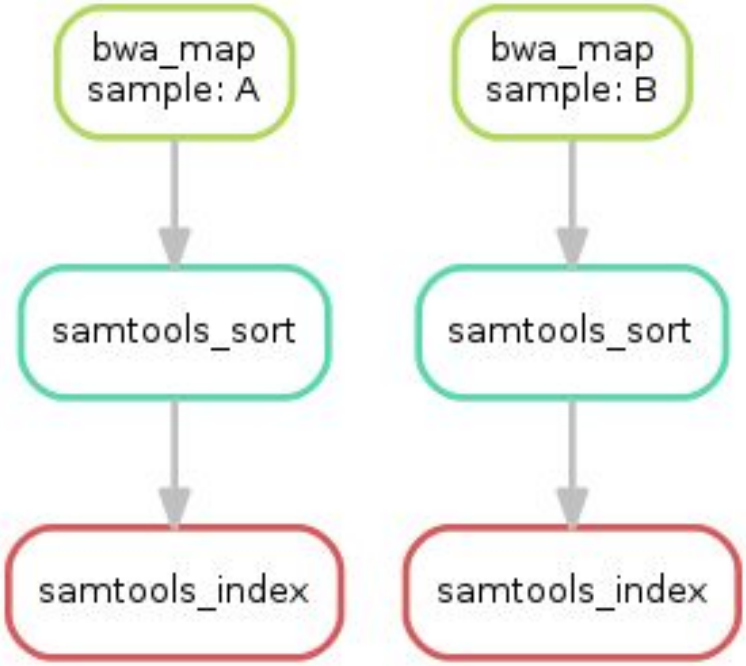
```
rule bwa_map:
    input:
        "data/genome.fa",
        "data/samples/A.fastq"
    output:
        "mapped_reads/A.bam"
    shell:
        "bwa mem {input} | samtools view -Sb - > {output}"
```

```
$snakemake -np mapped_reads/A.bam
```

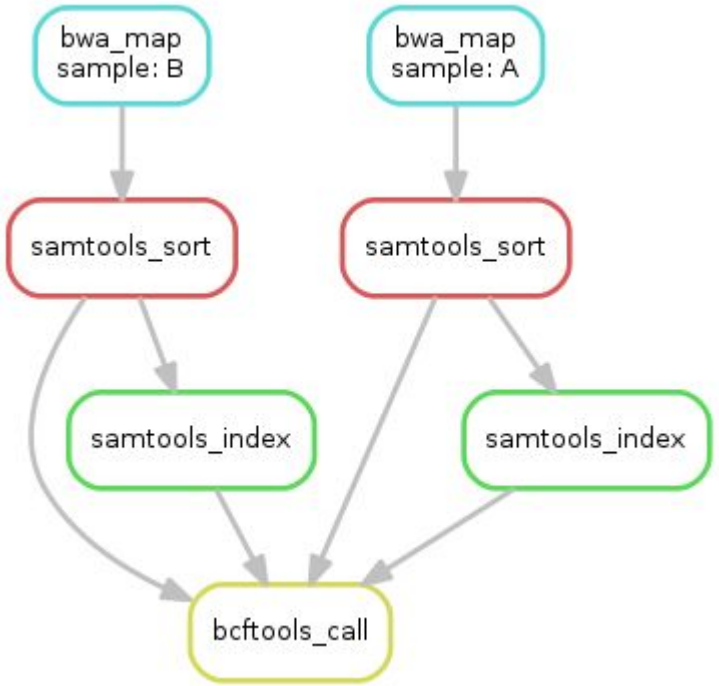
```
rule bwa_map:
    input:
        "data/genome.fa",
        "data/samples/{sample}.fastq"
    output:
        "mapped_reads/{sample}.bam"
    shell:
        "bwa mem {input} | samtools view -Sb - > {output}"
```

```
$snakemake -np mapped_reads/B.bam
```

```
rule all:
    input:
        "sorted_reads/A.bam.bai",
        "sorted_reads/B.bam.bai"


rule bwa_map:
    input:
        "data/genome.fa",
        "data/samples/{sample}.fastq"
    output:
        "mapped_reads/{sample}.bam"
    shell:
        "bwa mem {input} | samtools view -Sb - > {output}"


rule samtools_sort:
    input:
        "mapped_reads/{sample}.bam"
    output:
        "sorted_reads/{sample}.bam"
    shell:
        "samtools sort -T sorted_reads/{wildcards.sample} "
        "-O bam {input} > {output}"


rule samtools_index:
    input:
        "sorted_reads/{sample}.bam"
    output:
        "sorted_reads/{sample}.bam.bai"
    shell:
        "samtools index {input}"
```



```
SAMPLES = ["A", "B"]


rule all:
    input:
        "calls/all.vcf"


rule bwa_map:
    input:
        "data/genome.fa",
        "data/samples/{sample}.fastq"
    output:
        "mapped_reads/{sample}.bam"
    shell:
        "bwa mem {input} | samtools view -Sb - > {output}"


rule samtools_sort:
    input:
        "mapped_reads/{sample}.bam"
    output:
        "sorted_reads/{sample}.bam"
    shell:
        "samtools sort -T sorted_reads/{wildcards.sample} "
        "-O bam {input} > {output}"


rule samtools_index:
    input:
        "sorted_reads/{sample}.bam"
    output:
        "sorted_reads/{sample}.bam.bai"
    shell:
        "samtools index {input}"


rule bcftools_call:
    input:
        fa="data/genome.fa",
        bam=expand("sorted_reads/{sample}.bam", sample=SAMPLES),
        bai=expand("sorted_reads/{sample}.bam.bai", sample=SAMPLES)
    output:
        "calls/all.vcf"
    shell:
        "samtools mpileup -g -f {input.fa} {input.bam} | "
        "bcftools call -mv - > {output}"
```



Snakemake workflows

Snakemake workflow catalog

A comprehensive catalog of standards compliant, public, Snakemake workflows

Standardized usage 212 | All workflows 2676

Search

	Workflow	Description	Topics	QC	Stars	Watchers
Usage	snakemake-workflows/rna-seq-star-deseq2	RNA-seq workflow using STAR and DESeq2	snakemake, sciworkflows, reproducibility, gene-expression-analysis, deseq2	license MIT; last commit november 2023; linting failed; formatting passed	287	12
Usage	snakemake-workflows/dna-seq-gatk-variant-calling	This Snakemake pipeline implements the GATK best-practices workflow	reproducibility, snakemake, sciworkflows, genomic-variant-calling, gatk	license MIT; last commit may 2021; linting passed; formatting failed	207	9

https://github.com/snakemake-workflows/docs