Streamline boring bioinformatics with Genomics workflows

Renaud Van Damme

What’s a genomic workflow?

Genomic workflows

- Data analysis apps to retrieve information from datasets
- Huge parallelisation, creating numerous job over a cluster
- Mix of tools and scripts
- Complex configurations and dependencies interactions between the said tools

Genomic workflows

To reproduce a typical computational biology paper with minimal expertise it takes 280 Hours.
https://doi.org/10.1371/journal.pone.0080278

That include:

- Understanding the paper and material used
- Installation, set up
- Finding parameters
- Workflow validation
- computing

Reproducibility– Current issues

1. Access to the tools mentioned in the paper
2. Difficulty of installation
3. Version issues (machine and software)

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

What does nextflow in all that?

Nextflow challenges

- Reproducibility
- Portability
- Scalability
- Usability
- Consistency

Reproducibility

- Pipeline written in stone and version controlled (Git)
- Native support for containers and conda environment
- The application handling and the configuration/deployment are separated
- The only parameter to change are the ressources and environment

Portability

One installation to run them all:

Various platform compatibility:

- Google cloud
- AWS
- Azure
- Grid engine
- SLURM

Scalability

- Parallelisation
- Can run on local computer to an HPC-cluster or cloud

Usability

- One command run the whole pipeline
- Parallelization is automatic and scheduled based on the available ressources
- You can write your internal code in any language (bash, R, Python)
- The intermediary files are handled
- DSL2 allow reuse of code in other scripts
- Continuous checkpoints for resuming and expanding the pipelines

Consistency

- Version control (git, bitbucket, github, gitlab)
- Container
- Open source

Negative side

- Language is Groovy
- Flexibility means complexity
- Nitpicking details in scripts failure

Why container?

Container vs Virtual Machine

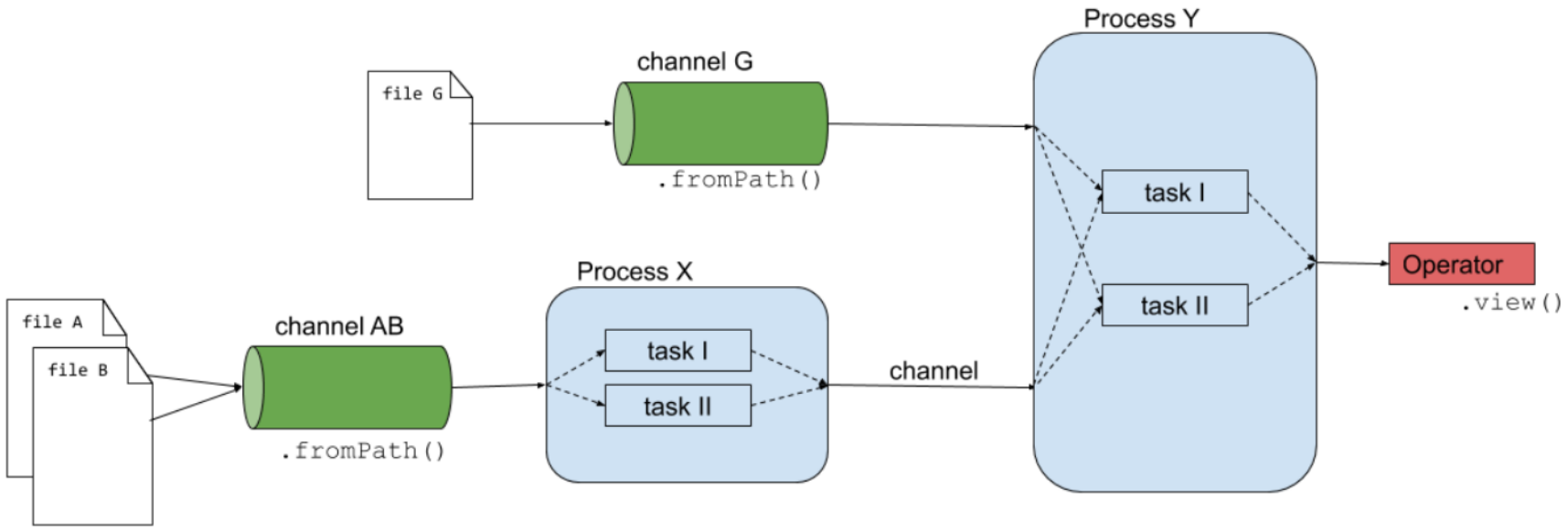
- Lighter: MB vs GB
- Faster startup as it virtualise a process and not an OS
- Immutable: do not change over time
- Composable: output of one can be feed directly in the next
- Transparent

Other workflow management system

Snakemake VS Nextflow

- CL oriented tool
- Rules defined using file name pattern
- Built-in support Singularity
- Custom script for cluster
- No support for source code management system
- Python based

- CL oriented tool
- Can manage any data structure
- Support all major container runtimes
- Built-in support for cluster and cloud
- Built-in support for Git/GitHub, manage pipeline revisions
- Groovy/JVM based



Source: https://github.com/vibbits/nextflow-jnj/blob/master/presentation/slidedeck.pdf

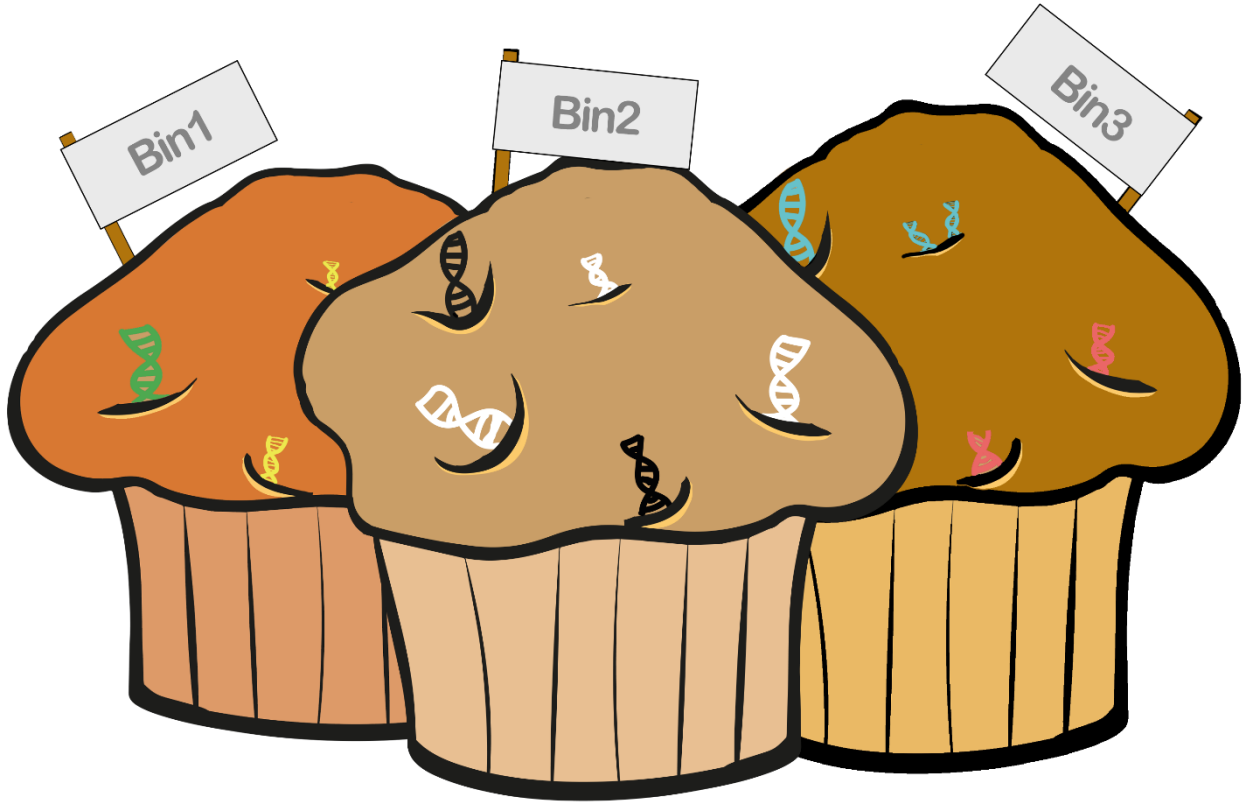
Conclusion

- Data analysis reproducibility is hard and underestimated
- Nextflow is not a magic wand but provide support for community and enables best-practices
- Splitting the application logic from the configuration enable self contained workflows
- Apps can be easily deployed across different environment with a single command
- The functional model allows apps to scale easily

Not enough?

- https://nf-co.re/
- https://github.com/nextflow-io/awesome-nextflow
- https://www.nextflow.io/
- https://www.nextflow.io/docs/latest/getstarted.html
- GitHub

Thanks for your attention



https://github.com/RVanDamme/MUFFIN/blob/master/.figure/Logo_MUFFIN_cropped.png

SCIENCE AND EDUCATION FOR SUSTAINABLE LIFE