Integrative Biology: Scientific workflows for computational reproducibility

Sarah Cohen-Boulakia

Université Paris-Sud, Laboratoire de Recherche en Informatique

CNRS UMR 8623, Université Paris-Saclay, Orsay, France

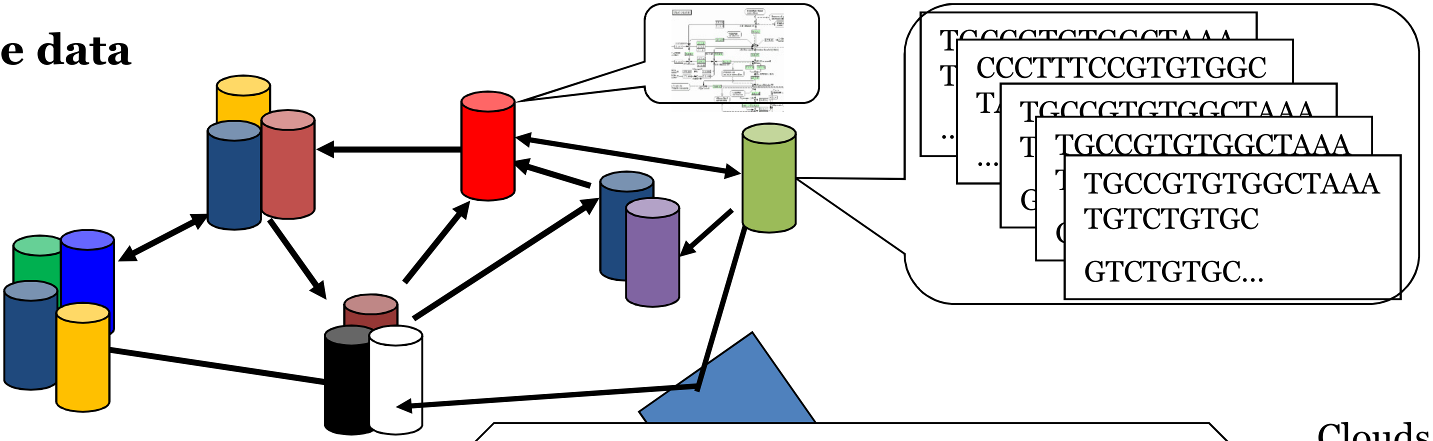
Bioinformatics analysis

Public and private data sources

Distributed

Heterogeneous

> 1,500



Binarization Water Use Efficiency

Segmentation

Java

Python

Web services

Clouds

Grids

Clusters

Desktop

Tools

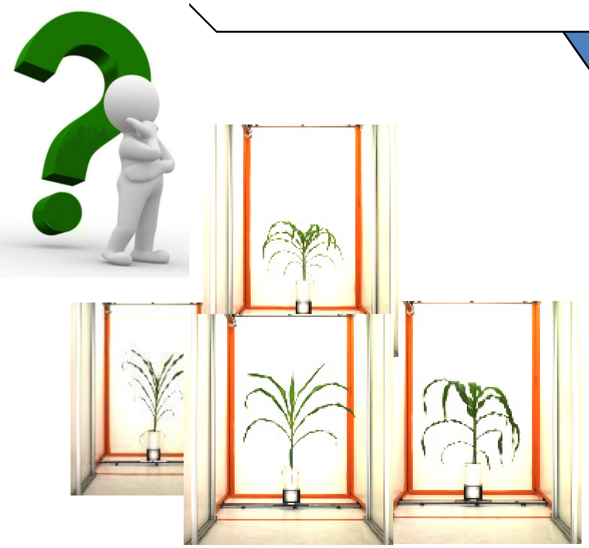
Distributed > 13,000

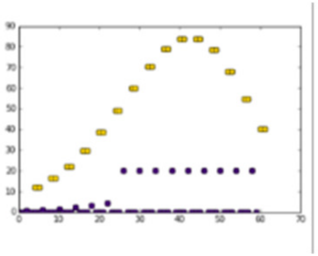
Heterogeneous

To be chained

How has this plot been generated?
With which input data?
With which tools?
Parameters?

→ **Reproducibility**





Biologist's workspace

Studies on reproducibility

- Nekrutenko & Taylor, Nature Genetics (2012)
  - 50 papers published in 2011 using the Burrows-Wheeler Aligner for Mapping Illumina reads.
  - 31/50 (62%) provide no information
    - no version of the tool + no parameters used + no exact genomic reference sequence
  - 7/50 (14%) provide all the necessary details
- Alsheikh-Ali et al, PLoS one (2011)
  - 10 papers in the top-50 IF journals → 500 papers (publishers)
    - 149 (30%) were not subject to any data availability policy (0% made their data available)
    - Of the remaining 351 papers
      - 208 papers (59%) did not adhere to the data availability instructions
      - 143 make a statement of *willingness* to share
      - 47 papers (9%) deposited full primary raw data online

Context, Challenges

Computational reproducibility crisis

Increasing number of irreproducible results

- Even published in high IF venues
- Not (always) deliberately
- Computational irreproducibility increases

Various scientific domains

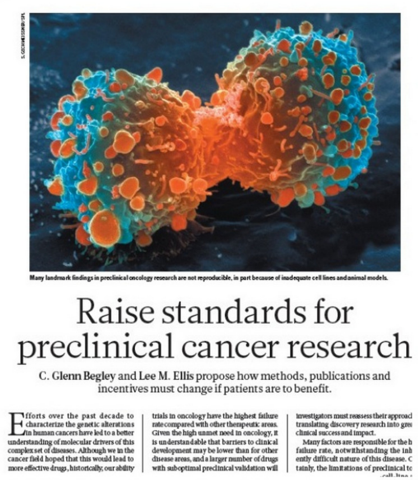
- Consequences may be huge (preclinical studies…)

Major challenge

- The cost of irreproducible preclinical studies have been evaluated to >$10 Billions per year (USA)

Becoming mandatory

- NSF projects, editors, ANR…



Many landmark findings in preclinical oncology research are not reproducible, in part because of inadequate cell lines and animal models.

Raise standards for preclinical cancer research

C. Glenn Begley and Lee M. Ellis propose how methods, publications and incentives must change if patients are to benefit.

Efforts over the past decade to characterize the genetic alterations in human cancers have led to a better understanding of molecular drivers of this complex set of diseases. Although we in the cancer field hoped that this would lead to more effective drugs, historically, our ability

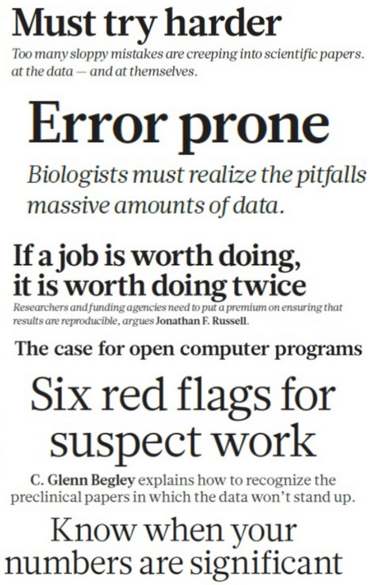
trials in oncology have the highest failure rate compared with other therapeutic areas. Given the high unmet need in oncology, it is understandable that barriers to clinical development may be lower than for other disease areas, and a larger number of drugs with suboptimal preclinical validation will

investigators must reassess their approac translating discovery research into gre clinical success and impact.

Many factors are responsible for the h failure rate, notwithstanding the inh ently difficult nature of this disease. C tainly, the limitations of preclinical to

47/53 "landmark" publications could not be replicated

[Begley, Ellis Nature, 483, 2012]



Must try harder

Too many sloppy mistakes are creeping into scientific papers. at the data — and at themselves.

Error prone

Biologists must realize the pitfalls massive amounts of data.

If a job is worth doing, it is worth doing twice

Researchers and funding agencies need to put a premium on ensuring that results are reproducible, argues Jonathan F. Russell.

The case for open computer programs

Six red flags for suspect work

C. Glenn Begley explains how to recognize the preclinical papers in which the data won't stand up.

Know when your numbers are significant

Reproducibility

V. Stodden *et al.*

The R Series

Implementing Reproducible Research

Edited by
Victoria Stodden
Friedrich Leisch
Roger D. Peng

CRC Press

A CHAPMAN & HALL BOOK

Empirical reproducibility

- detailed information about non-computational empirical scientific experiments and observations
- In practice this is enabled by making data freely available, as well as details of how the data was collected.

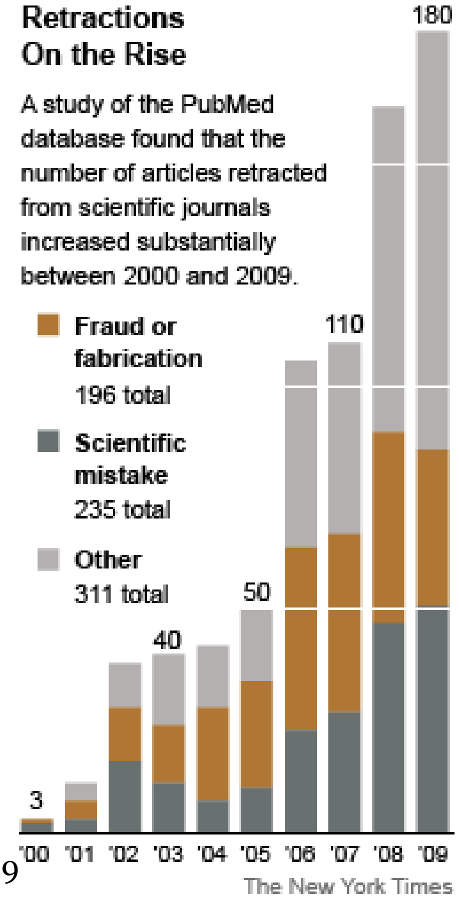
Statistical reproducibility

- detailed information about the choice of statistical tests, model parameters, threshold values, etc.
- This relates to pre-registration of study design to prevent p-value hacking and other manipulations.

Computational reproducibility

- detailed information about code, software, hardware and implementation details

→ Goal: document how data has been produced



Retractions On the Rise

A study of the PubMed database found that the number of articles retracted from scientific journals increased substantially between 2000 and 2009.

The New York Times

Scripts and reproducibility?
Good practices

Providing your scripts is an excellent first step
+ Using git/github for versioning, collaborative development

But scripts do not allow to

Distinguish between steps of the analysis

- piece of codes, methods/functions

… and execution of the analysis

- data sets used as inputs and then produced

Emphasize the major steps of the analysis

Provide solution for data management

- Naming convention for produced files, storage…

→ Scripts are difficult to share, exchange and reuse (repurpose)

Outline

Context

Scientific workflows

- Scientific workflow systems
- Companion tools

Lessons learnt on Scientific workflows and reproducibility

- Reprohackathons
- Levels of reproducibility with scientific workflows
- Reproducibility-friendly features

Open problems

Conclusion

Scientific workflow systems

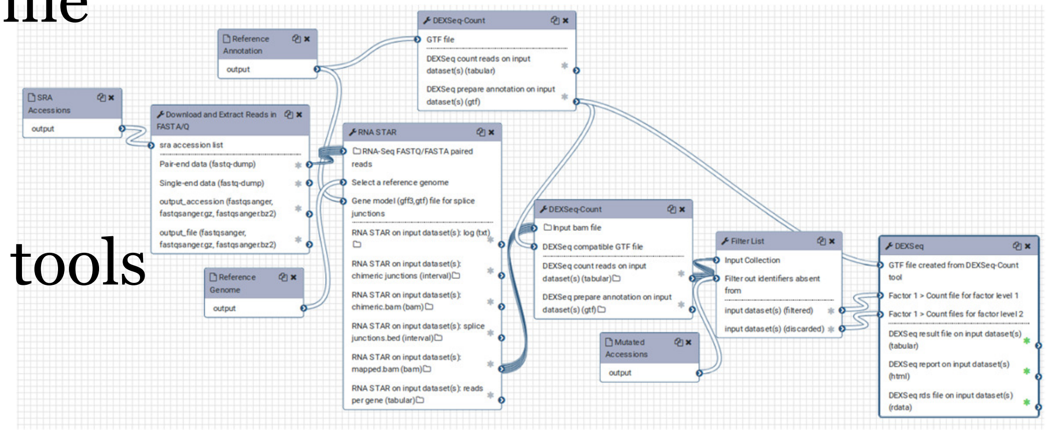
SWFS = “Data analysis pipeline”

Data flow driven

Encapsulation of scripts

WF specification: connected tools

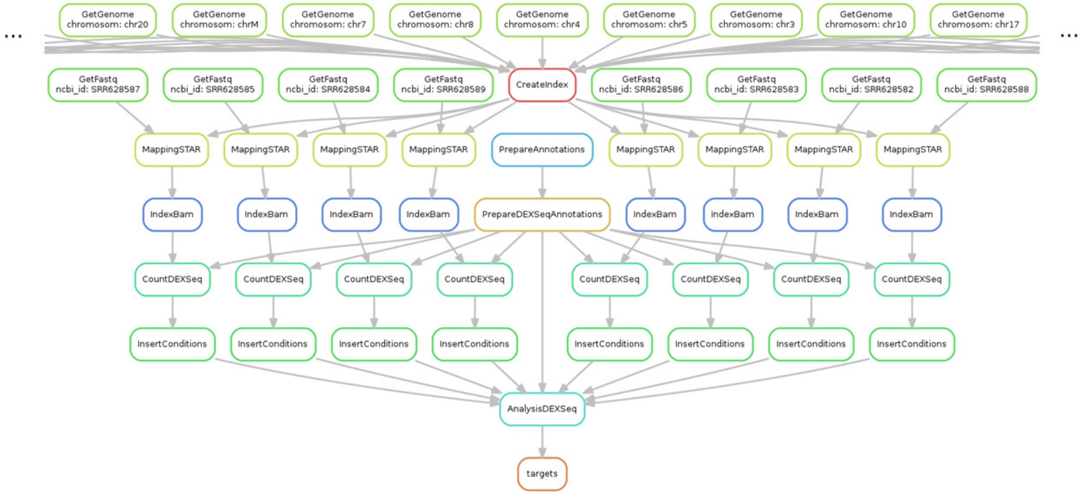
steps of the analysis


Reference Annotation
output
DEXSeq-Count
GTF file
DEXSeq count reads on input dataset(s) (tabular)
DEXSeq prepare annotation on input dataset(s) (gtf)
SRA Accessions
output
Download and Extract Reads in FASTA/Q
sra accession list
Pair-end data (fastq-dump)
Single-end data (fastq-dump)
output_accession (fastqsanger, fastqsanger.gz, fastqsanger.bz2)
output_file (fastqsanger, fastqsanger.gz, fastqsanger.bz2)
RNA STAR
RNA-Seq FASTQ/FASTA paired reads
Select a reference genome
Gene model (gff3,gtf) file for splice junctions
RNA STAR on input dataset(s): log (txt)
RNA STAR on input dataset(s): chimeric junctions (interval)
RNA STAR on input dataset(s): chimeric.bam (bam)
RNA STAR on input dataset(s): splice junctions.bed (interval)
RNA STAR on input dataset(s): mapped.bam (bam)
RNA STAR on input dataset(s): reads per gene (tabular)
Reference Genome
output
DEXSeq-Count
Input bam file
DEXSeq compatible GTF file
DEXSeq count reads on input dataset(s) (tabular)
DEXSeq prepare annotation on input dataset(s) (gtf)
Mutated Accessions
output
Filter List
Input Collection
Filter out identifiers absent from
input dataset(s) (filtered)
input dataset(s) (discarded)
DEXSeq
GTF file created from DEXSeq-Count tool
Factor 1 > Count file for factor level 1
Factor 1 > Count files for factor level 2
DEXSeq result file on input dataset(s) (tabular)
DEXSeq report on input dataset(s) (html)
DEXSeq rds file on input dataset(s) (rdata)

WF execution: data consumed/produced

Provenance modules

data management



SWFS scheduling, logging,

May be equipped with GUI

Galaxy, NextFlow, SnakeMake…

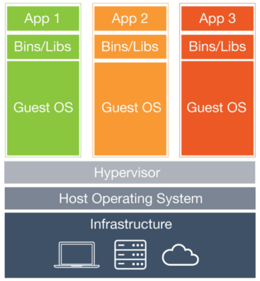
Capturing the programming environment

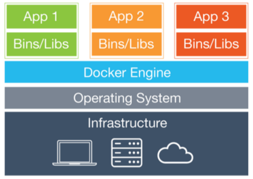
Ensuring your workflow has everything it needs to run

Libraries, dependencies…

Virtual machines capture the programming environment

Container solutions





- package an application
  - with all of its dependencies
  - into a standardized unit for software development

include the application and its dependencies

- but share the kernel with other containers
- They
  - are not tied to any specific infrastructure;
  - run on any computer, on any infrastructure and in any cloud

Lighter solution than classical VM

➔ BioContainers: a registry of containers!

Outline

Context

Scientific workflows

- Scientific workflow systems
- Companion tools

Lessons learnt on Scientific workflows and reproducibility

- Reprohackathons
- Levels of reproducibility with scientific workflows
- Reproducibility-friendly features

Open problems

Conclusion

Our new concept: ReproHackathon

Hackathon

- Several developers in the same room
- Same goal to achieve (e.g., predicting plants grow)
- Create useable software in a short amount of time
- Aim: Demonstrating feasibility

ReproHackathon

- A hackathon where
  - Given a scientific publication + input data (+ possibly contacts with authors)
  - Several (groups of) developers reimplement the methods to try to get the same result
- Aim: Ability of current workflow systems and companion tools to reproduce a scientific result

Editions of ReproHackathon

https://ifb-elixirfr.github.io/ReproHackathon/hackathon_1.html

First edition

- RNA-Seq data from patients with uveal melanoma: genes involved
- Divergent published results…
- 25 participants (IGRoussy, Curie, Pasteur, Saclay, Paris, Nantes, …)

Workflow Systems : SnakeMake, NextFlow, Galaxy…
Executed in the Cloud@IFB

+ Reprohackathon 2 in Lyon, July 2018
Phylogenetics

+ (coming) Reprohackathon 3
Montpellier Nov 25-27 2019
Plant phenotyping analysis

Outline

Context

Scientific workflows

- Scientific workflow systems
- Repositories of scientific workflows
- Companion tools

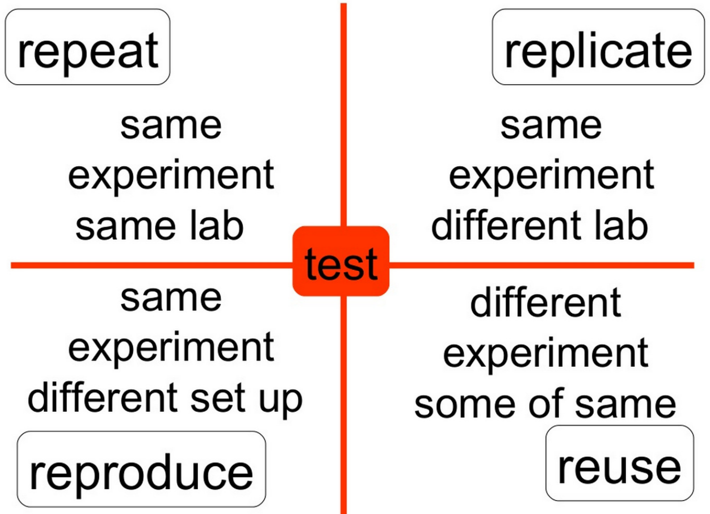
Lessons learnt on Scientific workflows and reproducibility

- Reprohackathons
- Levels of reproducibility with scientific workflows
- Reproducibility-friendly features

Open problems

Conclusion

Levels of computational reproducibility


repeat
same experiment same lab
replicate
same experiment different lab
test
same experiment different set up
reproduce
different experiment some of same
reuse

Drummond C Replicability is not Reproducibility: Nor is it Good Science, online
Peng RD, Reproducible Research in Computational Science *Science 2 Dec 2011: 1226-1227.*

3 ingredients

Workflow Specification

Chained Tools

Workflow Execution

Input data and parameters

Environment

OS/librairies …

Repeat

- *Redo*: exact same context
- Same workflow, execution setting, environement
- Identical *output*

→Aim = proof for reviewers ☺

Replicate

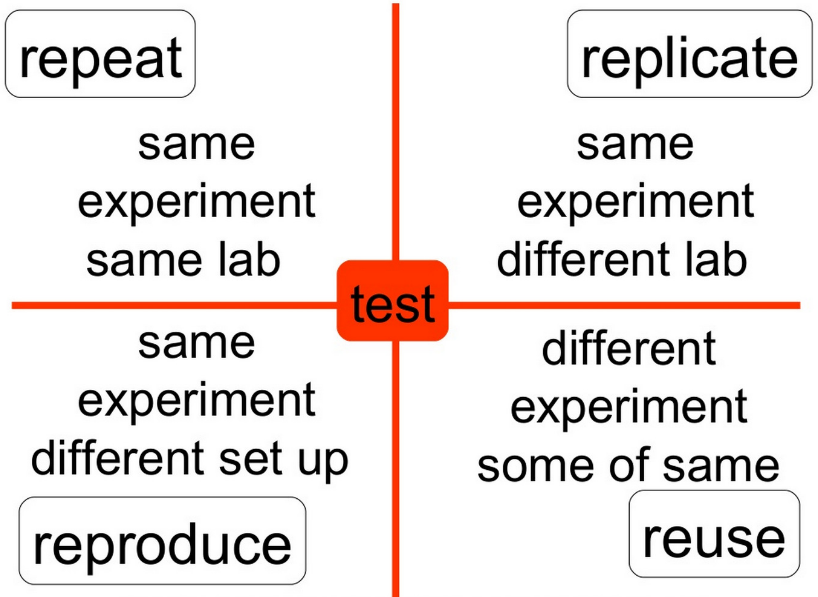
- Variation allowed in the workflows, execution setting, environement
- Similar *output*

→ Aim = robustness

A continuum of possibilities

Reproduce

- Same *scientific result*
- But the means used may be changed
- Different workflows, execution setting, environment
- Different output but in accordance with the result



repeat

same experiment same lab

replicate

same experiment different lab

test

same experiment different set up

reproduce

different experiment some of same

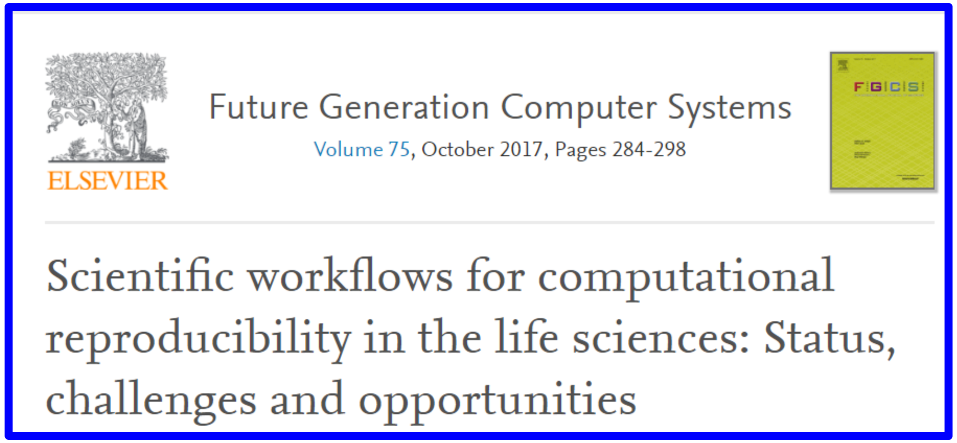
reuse

Drummond C Replicability is not Reproducibility: Nor is it Good Science, online
Peng RD, Reproducible Research in Computational Science *Science 2 Dec 2011: 1226-1227.*

Reuse

- Different scientific result
- Use of tools/… designed in another context

Reproducibility-friendly features



ELSEVIER

Future Generation Computer Systems

Volume 75, October 2017, Pages 284-298

Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities

6 Systems: Galaxy, Nextflow, SnakeMake, VisTrails, OpenAlea, Taverna

Specification

- Language (XML, Python…)
- Interoperability (CWL…)
- Description of steps
  - Remote services
  - Command line
  - Access to source code
- Modularity (nested workflows?)
- Annotation (tags, ontologies, myexperiment…)

Execution

- Language and standard (PROV…,) → repeat … reuse
- Presentation (interactivity with the results/provenance, notebooks) → replicate … reuse
- Annotations → reuse

Environment

Ability to run workflows within a given environment

- Virtual machines
  - VMWare, KVM, VirtualBox, Vagran,…
- Lighter solutions (containers)
  - Docker, Rocket, OpenVZ, LXC, Conda
- Capturing the command-line history, input/output, specification: CDE, ReproZip

Outline

Context

Scientific workflows

- Scientific workflow systems
- Repositories of scientific workflows
- Companion tools to ensuring properly rerun

Lessons learnt on Scientific workflows and reproducibility

- Reprohackathons
- Levels of reproducibility with scientific workflows
- Reproducibility-friendly features

Open problems

Conclusion

Developing workflows

Bridge the gap between scripts and workflows

Supporting several programming languages in the same environment of development

Tests in workflows

- Unit tests, integration tests…
- Providing samples may be an issue (privacy…)

Workflow Maintenance: set of compatible libraries?

- Docker, VM allows to freeze the environment

→ Need to liquefy!

- Given a program P that can be repeated in an environment E… … Find an environment E' (E' uses more recent versions of libraries than E) where P still *works*

Discovering workflows [Reuse]

Query languages for repositories?

- Given a input and/or and output format/type
- *Given a workflow – find similar workflows*

Core of the problem: Workflow similarity

- State-of-the-art [SCB+14]
- Based on the graph structures or annotations (ontologies)
- Need to design hybrid and efficient solutions

NB : Same point with Reproducible papers (Notebooks)

Efficiently reusing (and searching for) Notebooks is an open point

Workflow citation

- Give credit
- Workflow history (several workflows may be involved)

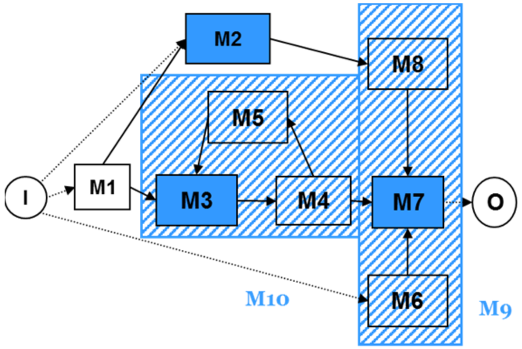
Simplifying workflows [Reuse]

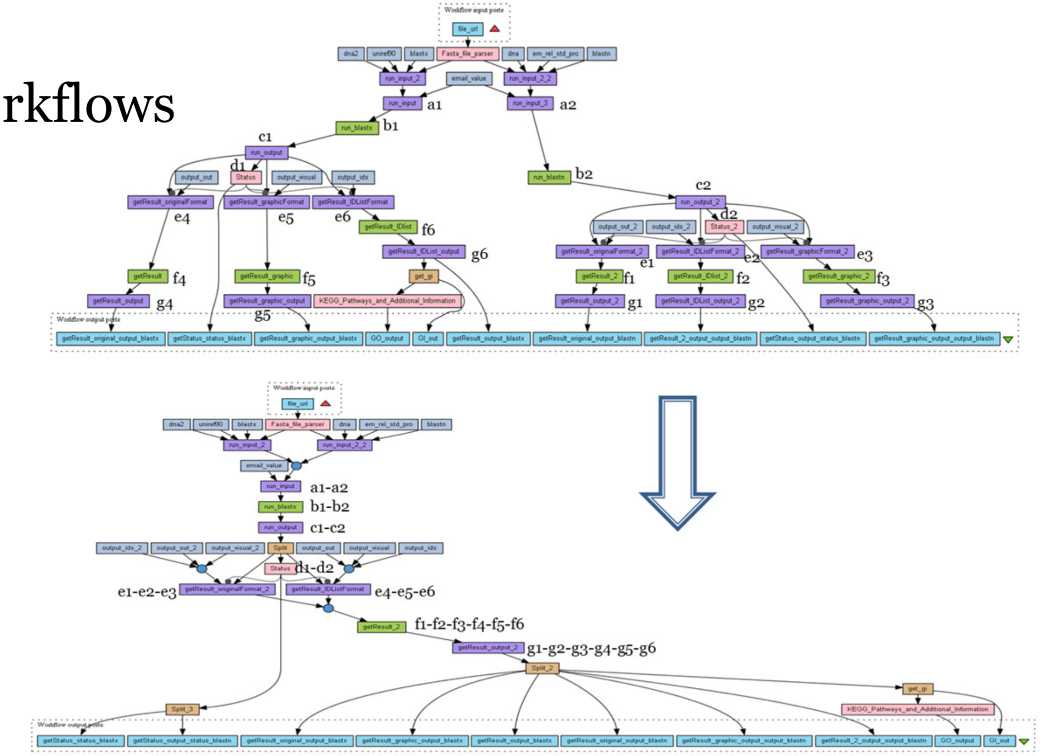
Designing more coarse-grained workflows

- Automatic Design of subworkflows (graph-based)
- Abstraction of provenance traces
- Summarization (Web Semantics)

Refactoring workflows

- Remove redundancies in workflows
- Rewritting, Anti-patterns





Conclusion

Many scientific results are not computationally reproducible

Providing scripts is an excellent start

Scientific workflows are increasingly mature solutions

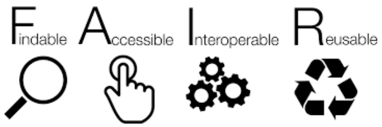
- Tracking the exact connected tools used
- Track the exact data used, produced and tool parameters setting

  → Provenance modules
- Coarse-grain version of the analysis to better capture the analysis steps

Several open challenges are directly related to improvement in research in computer science (graphs, algorithmics…)

Workflows play key role to produce FAIR data

FAIR metrics for workflows have to be defined too!



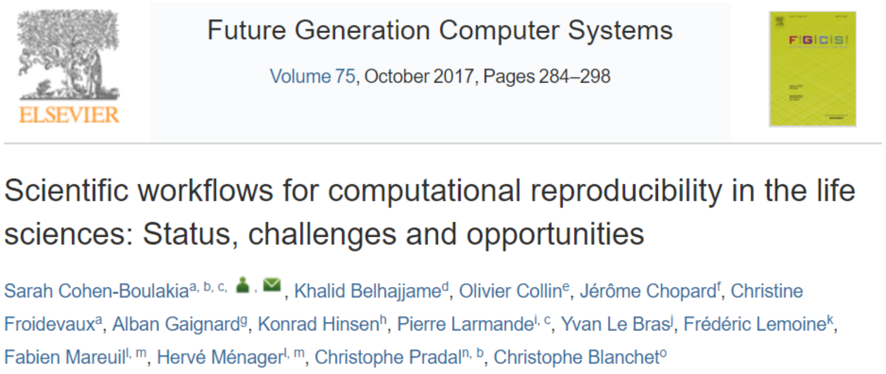
Results

(1) Paper @ FGCS

Levels of reproducibility

Criteria of choice

Open Challenges



Future Generation Computer Systems

Volume 75, October 2017, Pages 284–298

Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities

Sarah Cohen-Boulakia[a, b, c], Khalid Belhajjame[d], Olivier Collin[e], Jérôme Chopard[f], Christine Froidevaux[a], Alban Gaignard[g], Konrad Hinsen[h], Pierre Larmande[i, c], Yvan Le Bras[j], Frédéric Lemoine[k], Fabien Mareuil[l, m], Hervé Ménager[l, m], Christophe Pradal[n, b], Christophe Blanchet[o]

https://hal.archives-ouvertes.fr/hal-01516082/document

(2) 3 hour Webinar : Tutorial + 2 demos

(3) ReproHackathon

New concept designed

3 editions

- RNA seq 06/2017 Gif, PhiloData 07/2018, Lyon
- Next edition 25-27 Nov. 2019 Plant phenotyping, Montpellier

université
PARIS-SACLAY
CompBio

Join us!

cohen@lri.fr

PS: Bioinfo@LRI is hiring!