

Trends in Bio Data Integration: Scientific Workflows and Bio data ranking

Sarah Cohen-Boulakia

Université Paris Sud, LRI CNRS UMR 8623

<https://www.lri.fr/~cohen/BIGDATA/biodata-ami2b.html>

Introduction

- ▶ **Data Integration in the Life Science (DILS)** is more important than ever
- ▶ **Portals** are very popular
 - (+) Perform syntactic integration and keep the data in their original sources
 - (-) No semantic integration, data have to be inspected in each source
- ▶ **Data warehouses** remains the most frequently integration solution used in the Life Science community
 - (+) Semantic integration, huge computation is possible
 - (-) Data are copied, updating the warehouse may be highly difficult
- ▶ Biological Data are **Big Data: 5 V's**
 - **Volume**
 - **Velocity** (Data are obtained quickly and must be analyzed quickly)
 - **Variety** (Heterogeneous)
 - **Variability** (personalized medicine)
 - **Value** (Quality)...

... challenges are numerous...

Trends

- ▶ The complexity of the questions to be answered has increased a lot
 - **Integration requires analysis** and analysis requires integration
 - **Scientific workflows**
- ▶ The diversity of the sources has increased a lot
 - Inclusion of **quality** as a first-class citizen
 - **Ranking** of integrated search results
- ▶ The number of sources to be used has increased a lot
 - **Scalability** of integration in number of sources
 - One major goal of the **Semantic Web**, development of **ontologies**

This Tutorial

- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - Workflows and reproducibility
 - Current challenges

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

- ▶ **Part III – Conclusions**

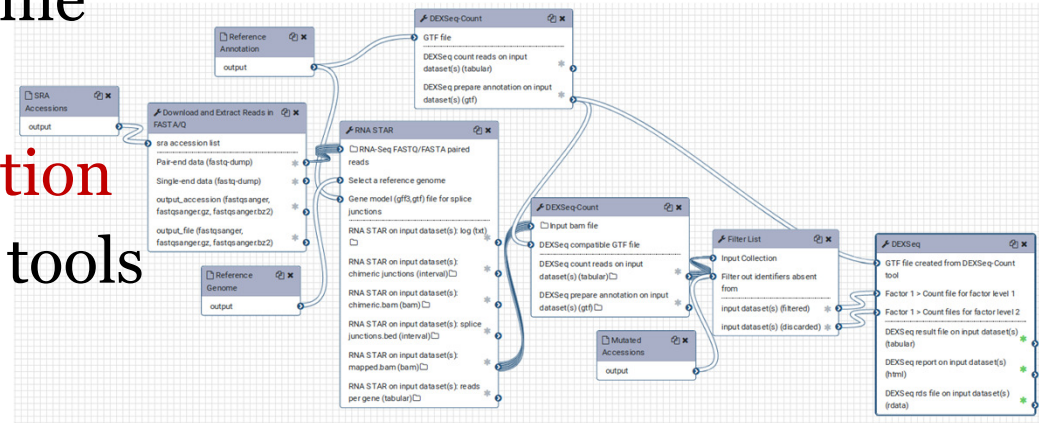
Scientific workflow systems (specification)

SWFS = “Data analysis pipeline”

Data flow driven

Encapsulation & Modularization

WF specification: connected tools
steps of the analysis



Encapsulation

Scripts are contained into boxes (steps)

Prog. Interface: input, parameters, output

Unified representation of steps

Modularization

Steps are independent of each others’

→ reusability

Scientific workflow systems (execution)

WF execution: data consumed/produced

Transparent, optimized, Traceable

SWFS scheduling, logging

Transparent

Able to run in any environments

Optimized

Able to run on different contexts (cluster, desktop, ...)

Traceable

Keep track of the data consumed & produced during the execution

Provenance modules → *data management*



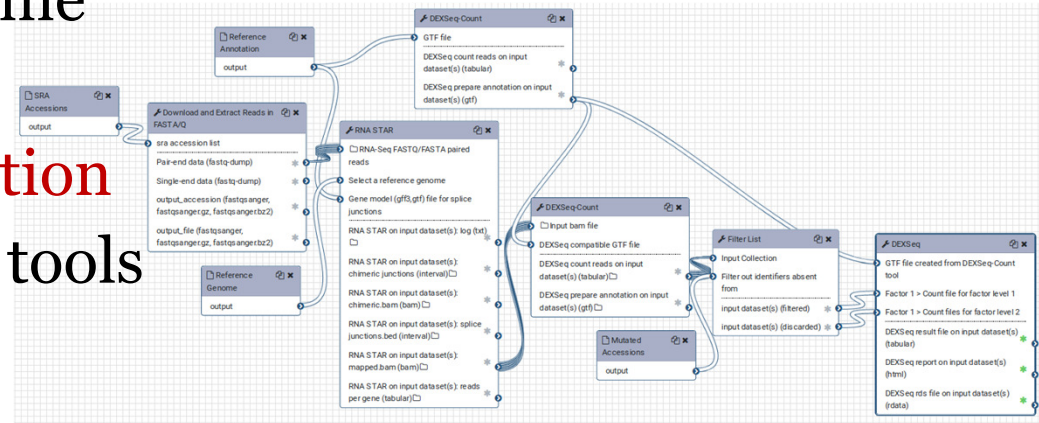
Scientific workflow systems (wrap-up)

SWFS = “Data analysis pipeline”

Data flow driven

Encapsulation & Modularisation

WF specification: connected tools
steps of the analysis



WF execution: data consumed/produced

Transparent, optimized, Traceable
data management

Mature systems: Galaxy, NextFlow, SnakeMake...



The Galaxy Project

- ▶ Galaxy is an **open source, web-based platform** for data intensive biomedical research.
- ▶ The **Galaxy Team** is a part of
 - the Center for Comparative Genomics and Bioinformatics at Penn State,
 - the Department of Biology and at Johns Hopkins University.
- ▶ The Galaxy Project is **supported** in part by
 - NSF,
 - NHGRI,
 - The Huck Institutes of the Life Sciences,
 - The Institute for CyberScience at Penn State,
 - and Johns Hopkins University...
- ▶ Can be used with
 - the **free public server** (usegalaxy.org)
 - or **other instances** (several in France: Institut Curie, **Institut Pasteur**, Genouest, SouthGreen...)

Galaxy main concepts

<https://wiki.galaxyproject.org/Learn>

- ▶ **Pages: documentation** within Galaxy. To supplement publications or to present tutorials.
- ▶ **Workflows**: define the **steps** in an analysis process. Workflows are analyses that are intended to be executed (one or more times) with different user-provided input Datasets. Steps come from the **toolshed**.
- ▶ **Histories** are analyses **records** in Galaxy that show all input, intermediate, and final datasets, as well as every step in the process and the settings used with each job executed.
- ▶ **Datasets** represent **individual files or jobs** included within a History.
- ▶ **Data Libraries** are collections of Datasets accessible. Designed for sharing datasets in between users or groups.


Workflow
specification

Workflow
execution

Designing and running a Galaxy workflow

- ▶ From scratch
 - Start with an input data set (type)
 - Drag-and-drop tools into the working environment
 - Connect tools (green means compatibility)
 - Parametrize tools
 - Upload a data set
 - Run the workflow on the data set
- ▶ Extract workflows from histories (reverse engineering)
- ▶ ...

Other major workflow systems

- ▶  Taverna <http://www.taverna.org.uk/>
 - Pioneer, Univ. Manchester
 - Perfect to combine [Web services](#)
 - Not used anymore
- ▶ **nextflow** <https://www.nextflow.io/>
 - Programmation-oriented (no GUI)
 - Increasingly used
 - Able to represent the specification with arcs labelled with data files names
- ▶ Snakemake <https://snakemake.readthedocs.io>
 - Programmation-oriented (no GUI)
 - Need to understand make commands ;)
 - The workflow is described as a set of rules
 - Ability to visualize the execution graph

And many others.... !

- ▶ Kepler (<https://kepler-project.org/>, BioKepler)
- ▶ Pegasus (<http://pegasus.isi.edu/>, Cloud ++)
- ▶ MobyLe (<http://mobyLe.pasteur.fr/>)
- ▶ OpenAlea (<http://openalea.gforge.inria.fr>, Plants ++)
- ▶ RapidMiner (<https://rapidminer.com/>)
- ▶ WINGS (<http://www.wings-workflows.org/>, semantics)
- ▶ KNIME (<https://www.knime.org/>)
- ▶ Cunieform (works on Hadoop YARN...)

Different systems for different users

▶ Snakemake & Nextflow

- + Excellent systems for programmers (prototyping)
- + Transparency, optimization of execution
- Impossible to be used by end-users
- Re-use, exchange /sharing

▶ Galaxy

- + Excellent system for end-users having admins ☺
- 2 kinds of users: programmers(admins) and end-users
- + Provides toolsheds containing tools already encapsulated
- end-users must use the tools available or ask admins
- + easy to share/exchange/reuse workflows within the same toolshed

This Tutorial

- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - **Repositories of workflows and web services (reuse)**
 - workflows and reproducibility
 - Current challenges

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

- ▶ **Part III – Conclusions**

Scientific Workflow Repositories



- Upload a scientific workflow
- Search, download & reuse existing scientific workflows
- Most specifically for single workflow system

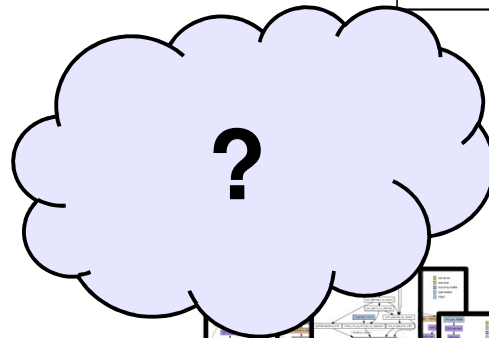
Scientific Workflow Discovery

Pose keyword query

Kepler

Search in textual annotations

my experiment



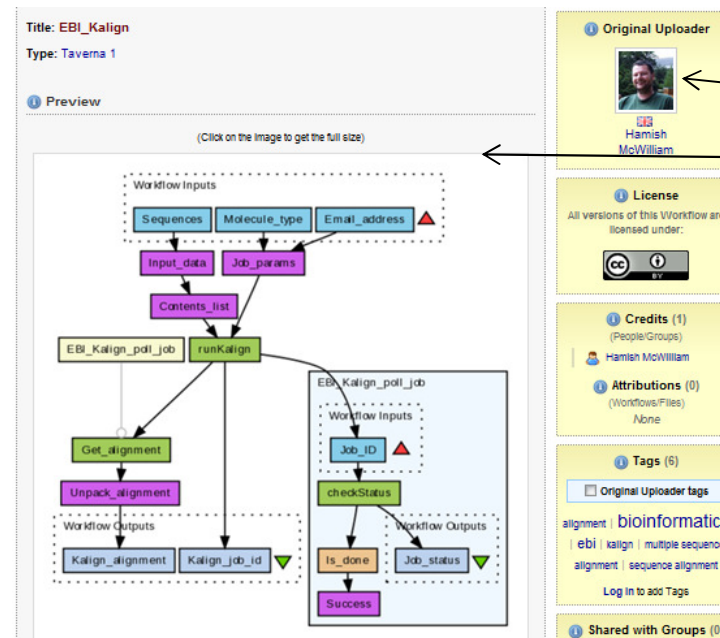
List of 10s or 100s of workflows

Reuse scientific workflow

Find appropriate workflows

myExperiment

- ▶ myExperiment.org
- ▶ Looking for workflows
 - By keywords
 - BioAID... workflow
 - Inspecting meta-data (author, favoured by, history...)
 - By authors
 - By group
 - ...



Conceptor
Workflow
Annotations
...

Bio.tools (replaces BioCatalogue)

<https://bio.tools/>

- ▶ Registry of **Tools** for the Life Sciences
 - find, understand, compare and select resources == **discovery**
 - use and connect them in workflows == **(inter)operability**
- ▶ Led by **ELIXIR** (European network of Excellence)
- ▶ Each tool must be described using **biotoolsSchema**
 - a formalized XML schema (XSD) which defines a description model for bioinformatics software (inputs, outputs and operations)
 - EDAM Ontology Terms are used
- ▶ **EDAM** Ontology
 - bioinformatics types of data including identifiers, data formats, operations and topics

Description of Tools in Bio.Tools

elixir Search tool and data services registry Login Register

BLAST API (EBI)

Sequence analysis >

Web API

NCBI BLAST is a sequence similarity search program.
http://www.ebi.ac.uk/Tools/webservices/services/sss/ncbi_blast_rest

Sequence comparison >

Publications

Primary
DOI >

Credits

BioCatalogue | Project
Documentor | Link >

Documentation

General >

67

- Blogged by 5
- Referenced in 2 policy sources
- Tweeted by 23
- Mentioned by 1 peer review sites
- On 1 Facebook pages
- Referenced in 15 Wikipedia pages
- Mentioned in 2 Q&A threads
- 1633 readers on Mendeley
- 25 readers on CiteULike

[See more details](#) | [Close this](#)

In BioCatalogue...

Looking for services given an input

Home » Search By Data

Search By Data

Paste in your input/output data and we will find all the operations that you can potentially use to process/analyse it...

```
>sp|P38398|BRCA1_HUMAN Breast cancer type 1 susceptibility protein
OS=Homo sapiens GN=BRCA1 PE=1 SV=2
MDLSALRVEEVQNVINAMQKILECPICLELIKEPVSTKCDHIFCKFCMLKLLNQ
KKGPSQ
CPLCKNDITKRSLQESTRFSQVEELLKIICAFQLDTGLEIANSYNFAKKENN
SPEHLKD
EVSIIQSMGYRNRARLLQSEPNPSLQETSLSVQLSNLGTVRTLRKQRIQ
PQKTSVYI
ELGSDSSEDVTKATYCSVGDQELLQITPQGTREISLDSAKKAACEFSETD
VTNTEHHQ
```

Maximum number of results:

Search data against:

- Service Inputs
- Service Outputs

Go!

PLEASE NOTE: Your searches might be logged and used for analysis and/or annotation of services in the future. Therefore do not use any secure or confidential data here.

Help us improve these results by adding example input/output data to existing services

TFmodeller SOAP 

Operation: [TFmodeller](#) | Port: [proteinFASTAsequence](#)

Example data matched:



by  Fr

```
>P0A9E5|FNR ECOLI Fumarate and nitrate reduction regulato...
MIPEKRIIRRIQSGGCAIHCQDCSISQLCIPFTLNEHELDQLDNI IERKKPIQKGQTLFK
AGDELKSLYAIRSGTIKSYTITEQGDEQITGFHLAGDLVGFDAIGSGHHPFAQALETSM
VCEIPFETLDDLSGRMPNLRQQMMRLMSGEIKGDQDMILLSSKKNAEERLAAFIYNLSRR
FAQRGFSPREFRLTMRGDIGNYLGLTVETISRLLGRFQKSGMLAVKGYIT IENNDALA
QLAGHTRNVA
```

TFmodeller SOAP 

Operation: [TFmodeller](#) | Port: [proteinFASTAsequence](#)

Example data matched:



```
>P0A9E5|FNR ECOLI Fumarate and nitrate reduction regulato...
MIPEKRIIRRIQSGGCAIHCQDCSISQLCIPFTLNEHELDQLDNI IERKKPIQKGQTLFK
AGDELKSLYAIRSGTIKSYTITEQGDEQITGFHLAGDLVGFDAIGSGHHPFAQALETSM
VCEIPFETLDDLSGRMPNLRQQMMRLMSGEIKGDQDMILLSSKKNAEERLAAFIYNLSRR
FAQRGFSPREFRLTMRGDIGNYLGLTVETISRLLGRFQKSGMLAVKGYIT IENNDALA
QLAGHTRNVA
```

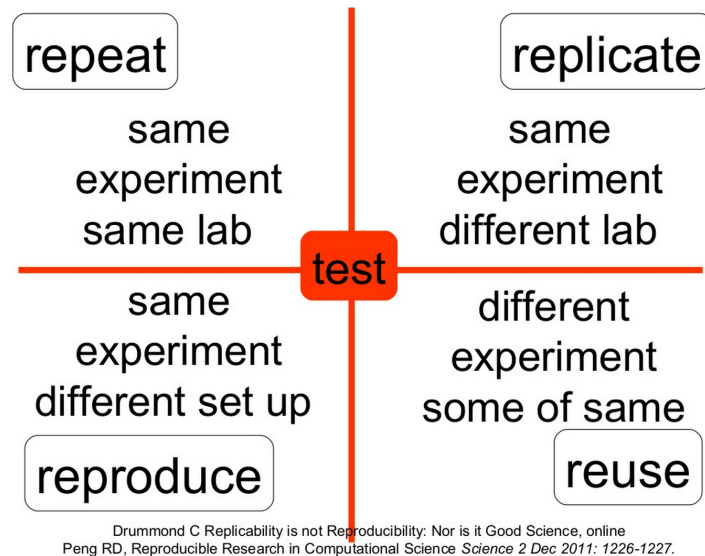
This Tutorial

- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - **Workflows and reproducibility**
 - Current challenges

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

- ▶ **Part III – Conclusions**

Repro with Workflows: ingredients and levels



3 ingredients

Workflows Specification

Chained Tools

Workflow Execution

Input data and parameters

Workflow Environment

OS/libraries ...

▶ Repeat

- *Redo*: exact same context
 - Same workflow, execution setting, environment
 - Same *output*
- Aim = proof for reviewers 😊

▶ Replicate

- Variation allowed in the workflows, execution setting, environment
 - Similar *output*
- Aim = robustness

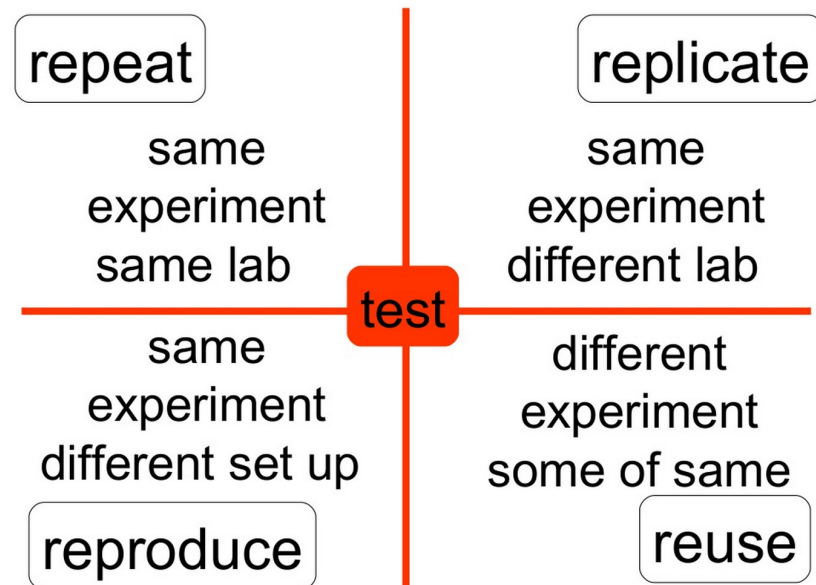
A continuum of possibilities

▶ Reproduce

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

▶ Reuse

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



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.

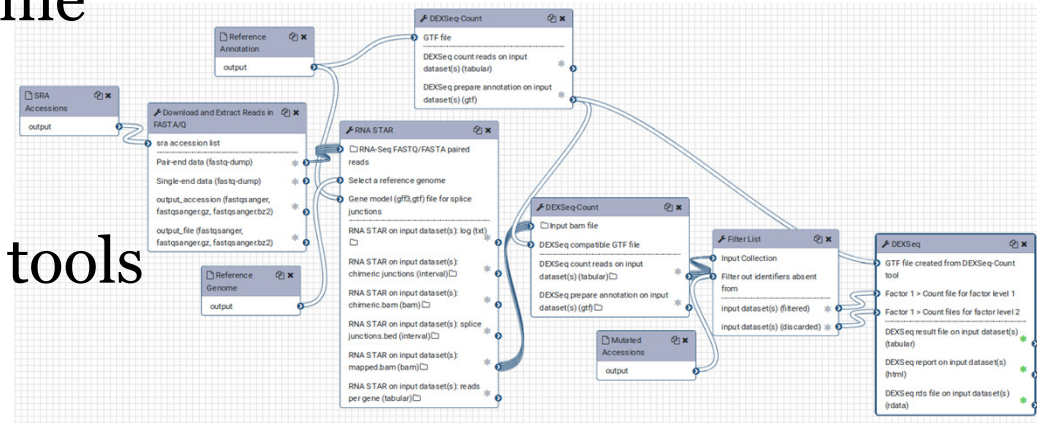
Scientific workflow systems (reminder)

SWFS = “Data analysis pipeline”

Data flow driven

Encapsulation of scripts

WF specification: connected tools
steps of the analysis



WF execution: data consumed/produced

Provenance modules

data management

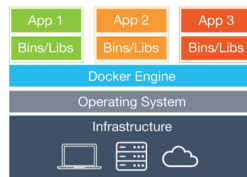
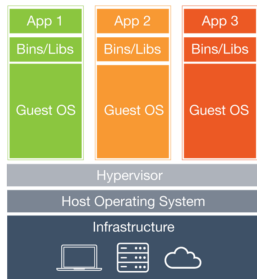
SWFS scheduling, logging,
May be equipped with GUI
Galaxy, NextFlow, SnakeMake...



Capturing the programming environment

Ensuring your workflow has everything it needs to run
Libraries, dependencies... → *Transparent execution*

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!**

Reproducibility-friendly features

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

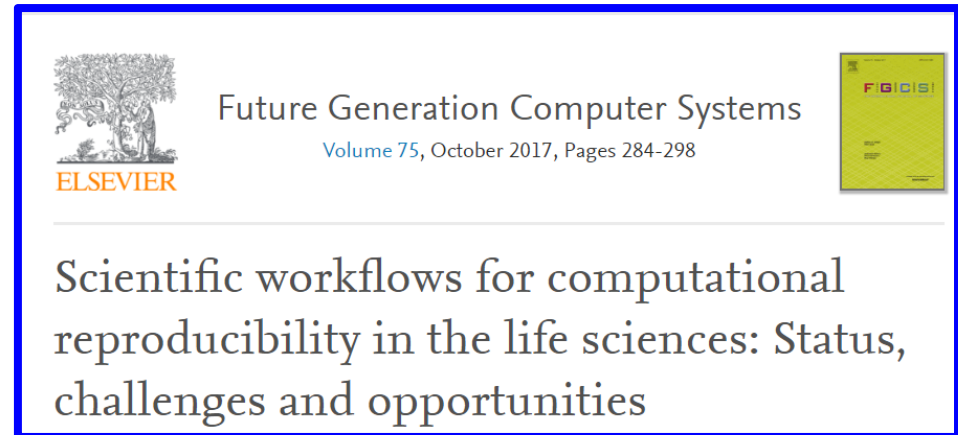
Ability to run workflows within a given environment

Virtual machines

- VMWare, KVM, VirtualBox, Vagrant,...

Lighter solutions (containers)

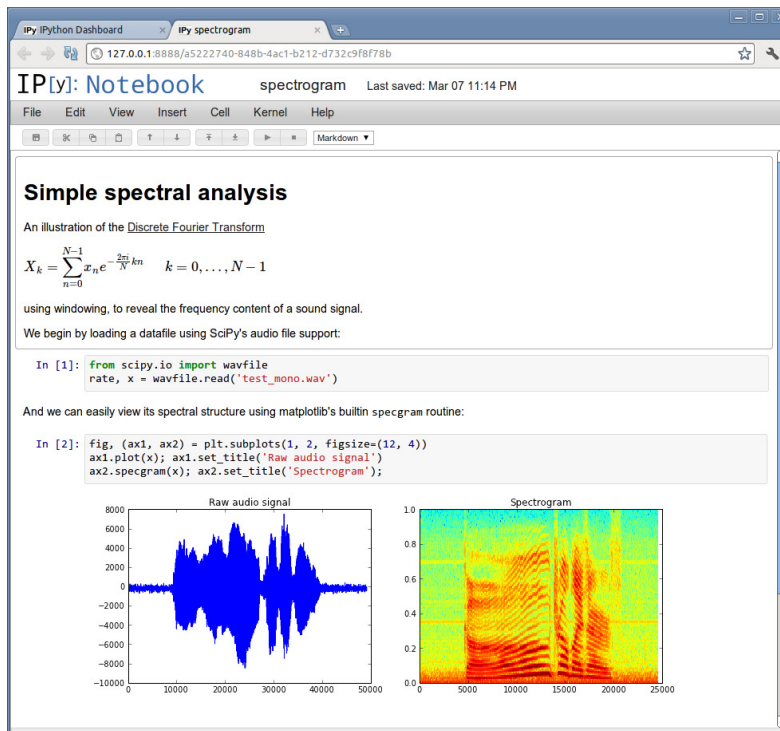
- Docker, Rocket, OpenVZ, LXC, Conda





Another kind of systems: Notebooks

- ▶ Web-based **interactive computational environment**
- ▶ Combination of code execution, text, mathematics, plots and rich media **into a single document**
- ▶ Some systems export workflow execution as executable Jupyter papers...



Excellent mean to explain/present a scientific results obtained

Ten Simple Rules for Reproducible Computational Research (PlosOne)

- ▶ 1: For Every Result, **Keep Track** of How It Was Produced
- ▶ 2: **Avoid Manual** Data Manipulation Steps
- ▶ 3: **Archive** the Exact Versions of All External Programs Used
- ▶ 4: **Version Control** All Custom Scripts
- ▶ 5: Record **All Intermediate Results**, When Possible in Standardized Formats
- ▶ 6: For Analyses That Include Randomness, **Note Underlying Random Seeds**
- ▶ 7: Always **Store Raw Data** behind Plots
- ▶ 8: Generate Hierarchical Analysis Output, **Allowing Layers of Increasing Detail to Be Inspected**
- ▶ 9: **Connect** Textual Statements to Underlying Results
- ▶ 10: **Provide Public Access** to Scripts, Runs, and Results

→ Several ways to follow them

→ More or less complex (from manually to fully automatically)

→ More or less time-consuming (repeat, reproduce,, reuse)

Wrap up

- ▶ Data Integration & Data Analysis
- ▶ Scientific workflows plays a major role to analyse bio data sets
- ▶ Major systems in place, large variety of solutions: Galaxy (GUI), SnakeMake/NextFlow (scripts)...
- ▶ Reproducibility and reuse is improved using such systems
 - **Specification**: which tools in what order
 - **Execution**: which data produced/consumed, which parameters
 - **Environment**: which OS, which librairies, ...
- ▶ Notebooks are another very interesting solution (to expose/explain a scientific result)

This Tutorial

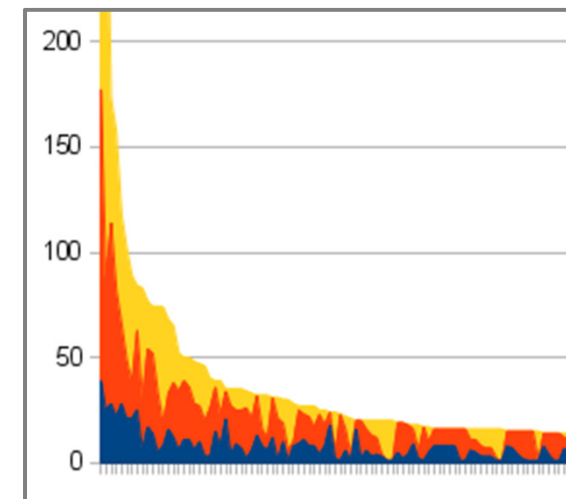
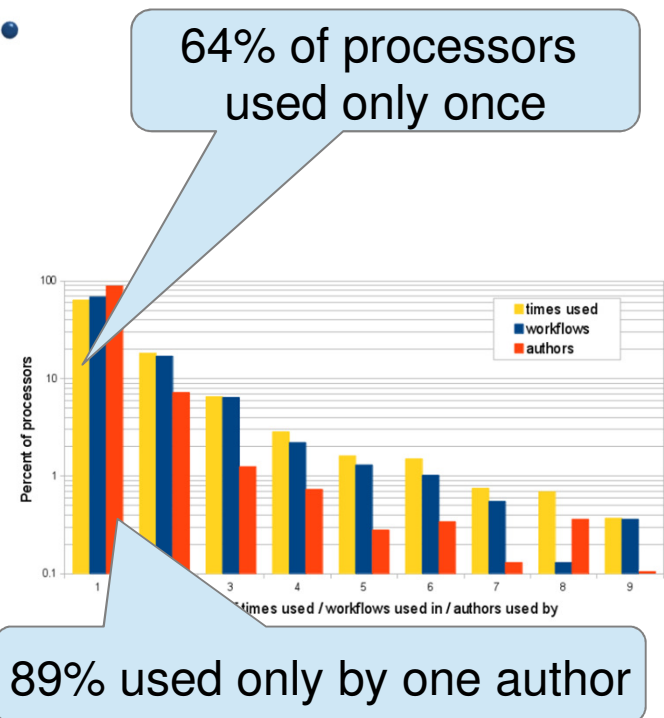
- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - workflows and reproducibility
 - [Latest results on workflows](#)
 - [Or How CS research may have direct impact on LS](#)

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

- ▶ **Part III – Conclusions**

Study on workflow reuse....

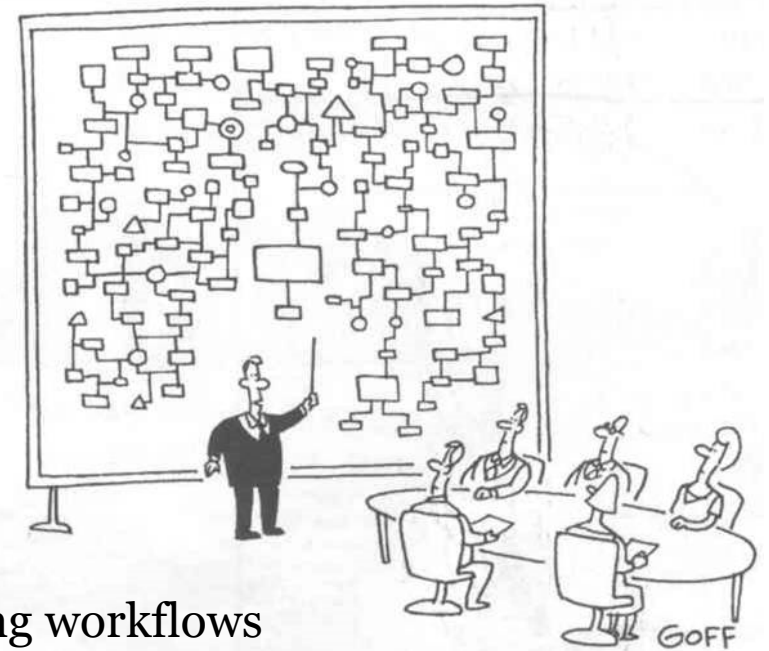
- **36% of elements are re-used**
 - These connect workflows quite densely
 - Can be exploited for repository IR
- Re-use rates have a **Zipf-like distrib**
 - **Local** : High re-use rates as-is
 - **Web-Service** : Authors have favorite services, unshared
 - **Script & subworkflows** : Authors have personal libraries
- **True cross-author re-use is low: 3%**
 - Authors have personal preferences & libraries
 - But don't use content from others



How to improve reuse?

Help finding
similar
workflows

Make
workflow
structures
less complex!



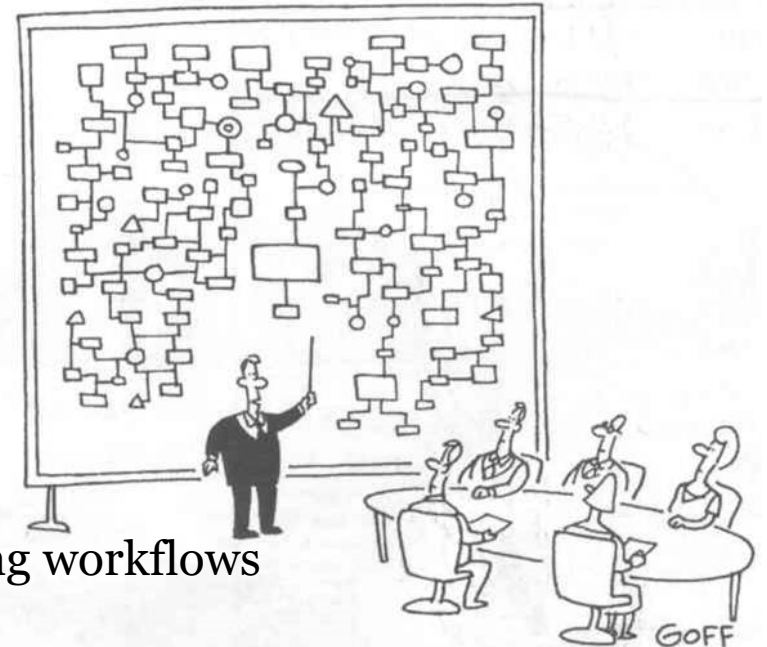
Plumbing workflows

How to improve reuse?

Help finding
similar
workflows

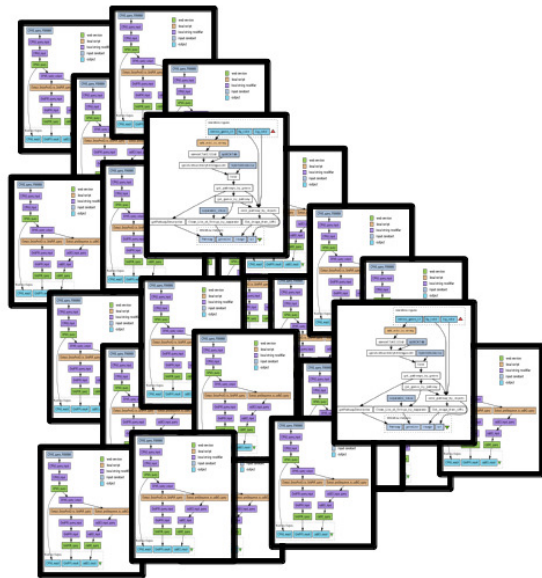


Make
workflow
structures
less complex!



Plumbing workflows

Scientific Workflow Discovery Improvement



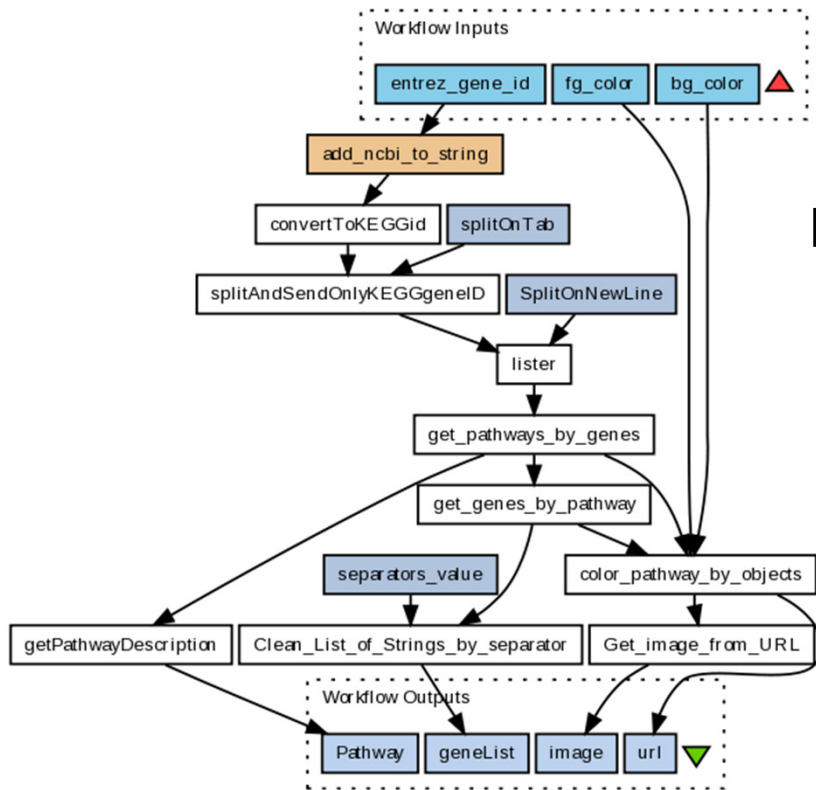
List of 10s or
100s of workflows

Goal

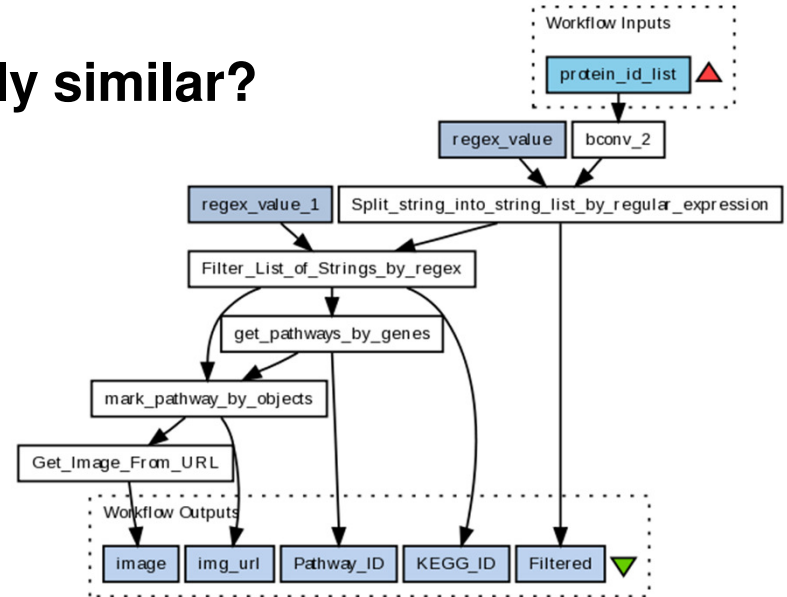
- **Group results** by similar workflows
- **Search by sample workflow**
- **Provide recommendations**
 - Similar workflows
 - Replacements
 - Extensions
- ...

Need: **Similarity Measures**

The Central Question

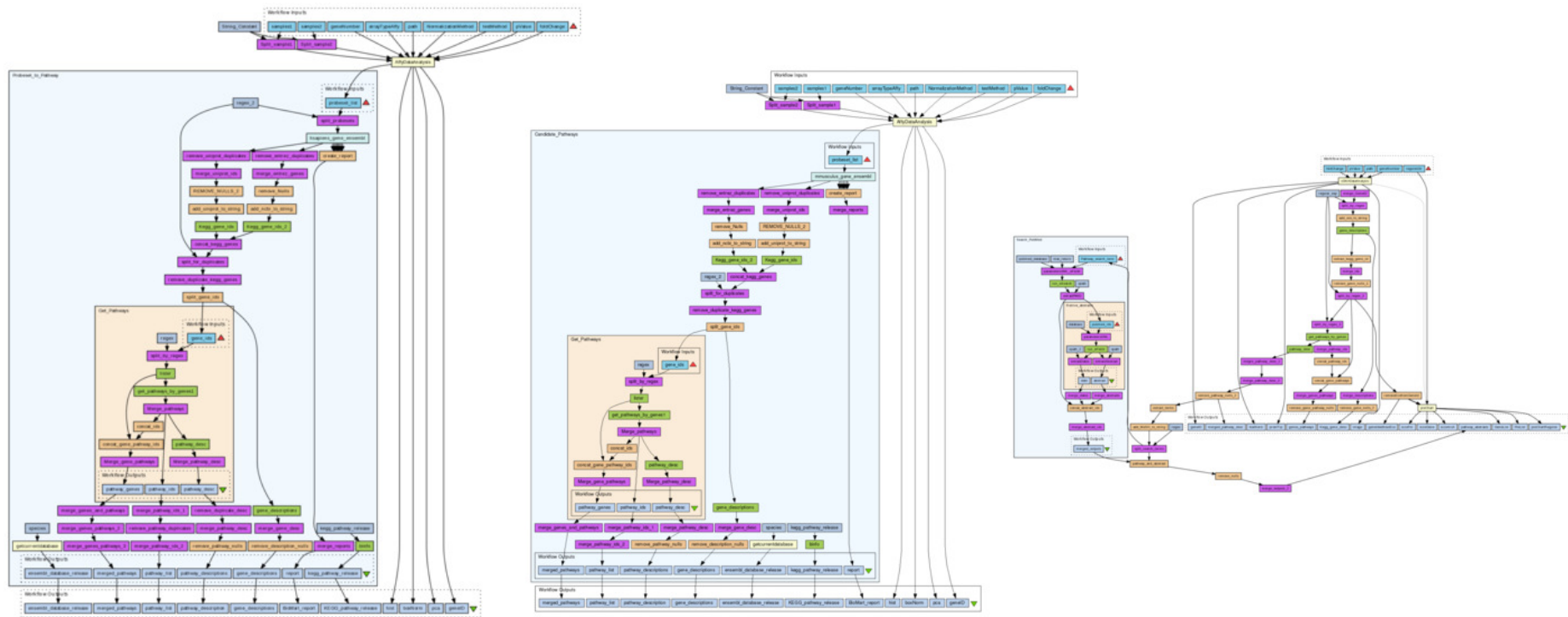


Functionally similar?



Example

workflows perform microarray analysis integrating various sources (pathway DB, probe mapping, PubMed)



- ▶ All three workflows may be used
 - entirely (which fits best?) or partly (from probes to pathways)

Similarity search for scientific workflows

[VLDB 2014]

With Johannes Starlinger,
Bryan Brancotte, Ulf Leser

► Framework

- capture all the sim. search techniques

- **Structure-based**

- Graph struct. of the workflow

- **Annotation-based**

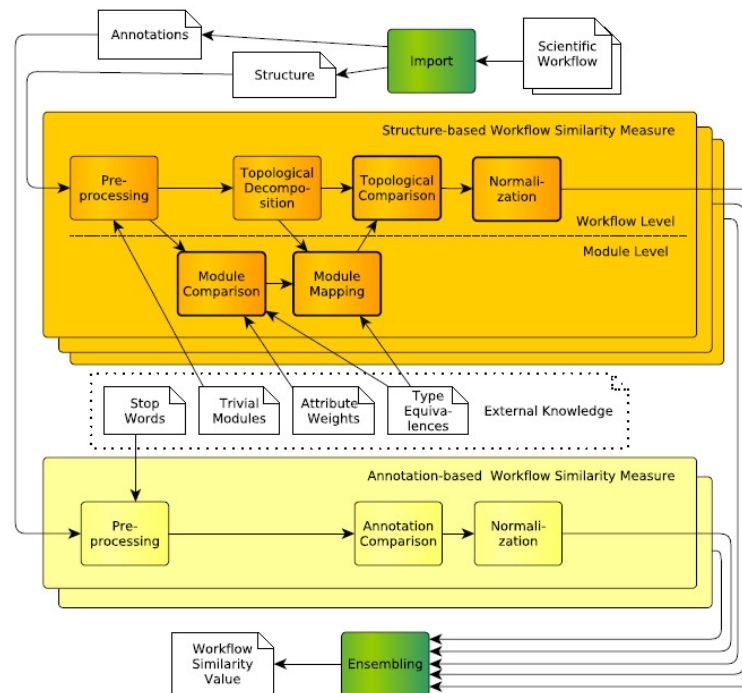
- Meta-data (description, tags...)

► Goal of the study

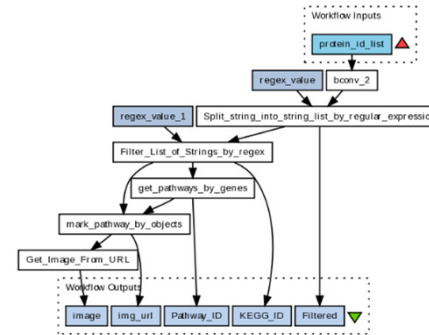
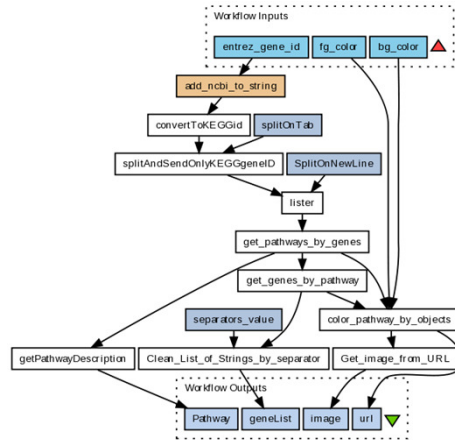
- compare results obtained by all techniques

- **On various data sets**

- Taverna, Galaxy, VisTrails

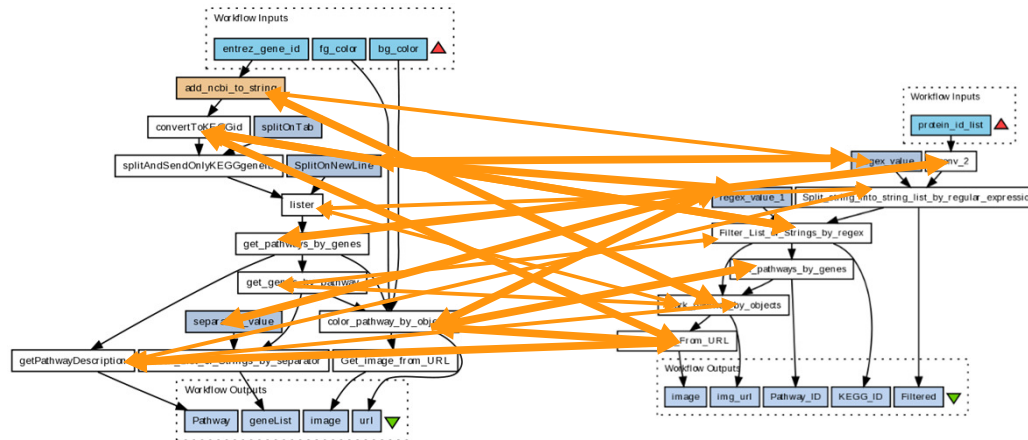


Subtasks of Scientific Workflow Comparison



Module Comparison

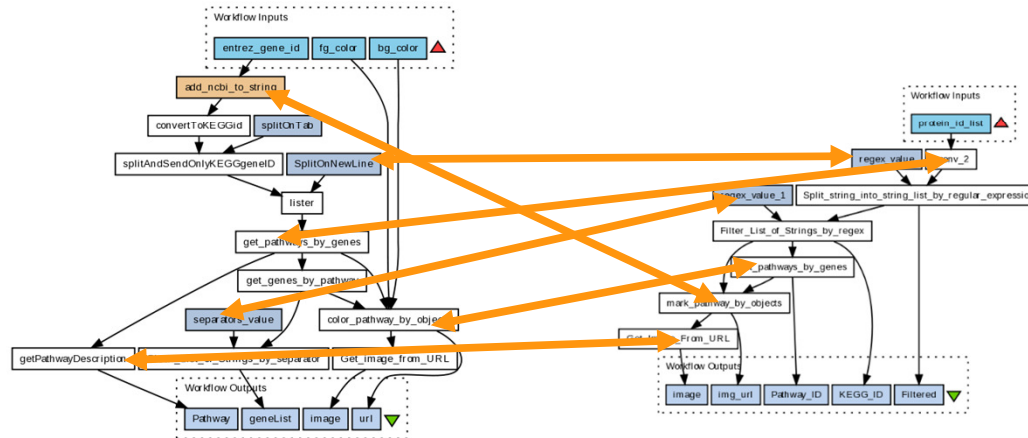
Subtasks of Scientific Workflow Comparison



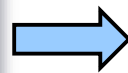
Module
Comparison

- Label
- Webservice Uri
- Scripts
- etc

Subtasks of Scientific Workflow Comparison



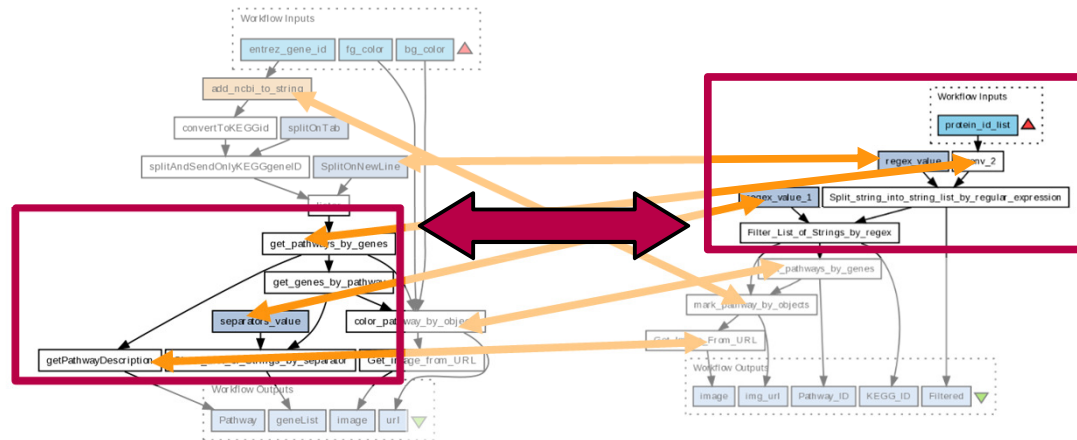
Module
Comparison



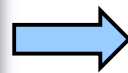
Module
Mapping

- greedy
- maximum weight

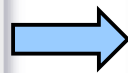
Subtasks of Scientific Workflow Comparison



Module Comparison



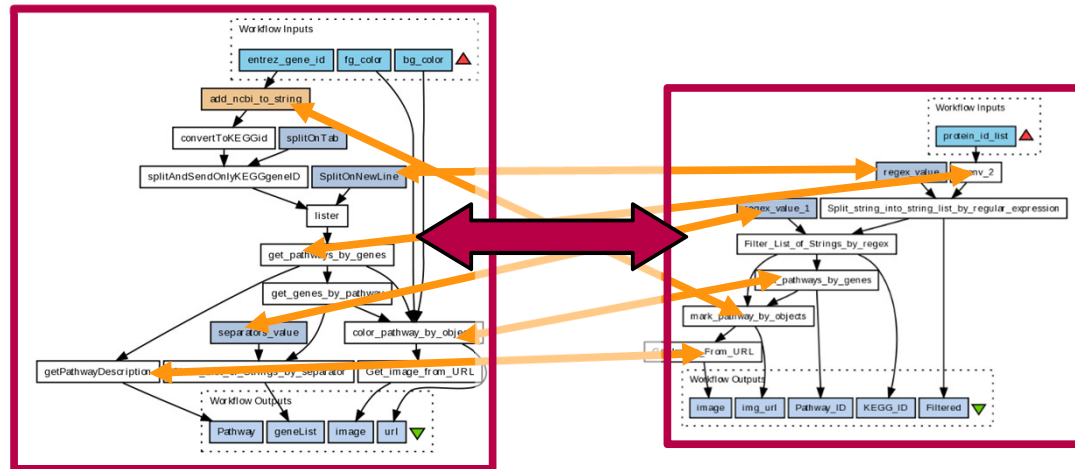
Module Mapping



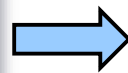
Topological Comparison

- Set of Modules
- Substructures
- Full Structure

Subtasks of Scientific Workflow Comparison



Module Comparison



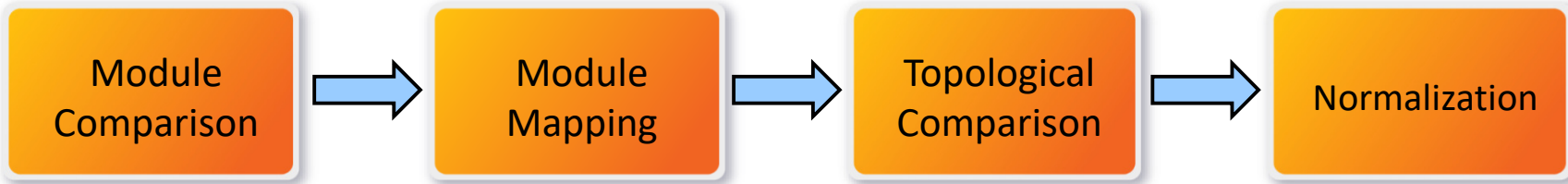
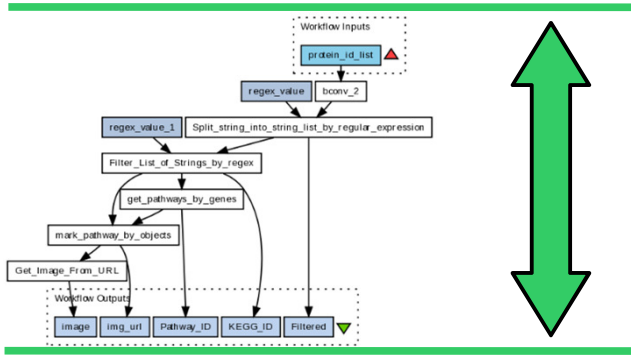
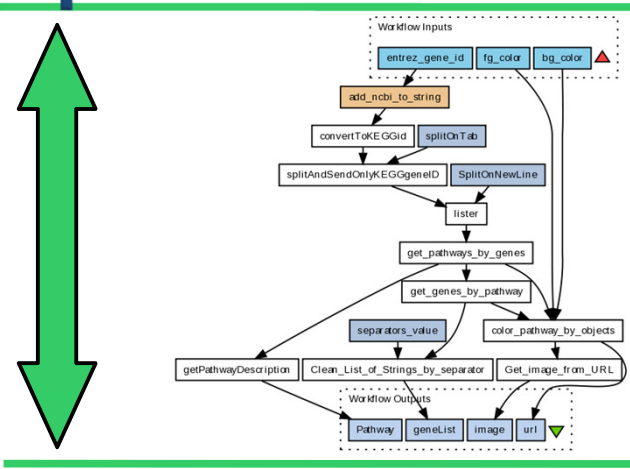
Module Mapping



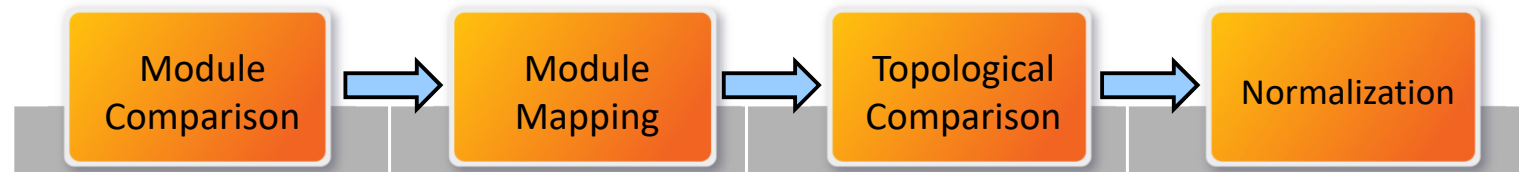
Topological Comparison

- Set of Modules
- Substructures
- Full Structure

Subtasks of Scientific Workflow Comparison



Existing Approaches



Stoyanovich et al.

Silva et al.

Bergmann et al.

Santos et al.

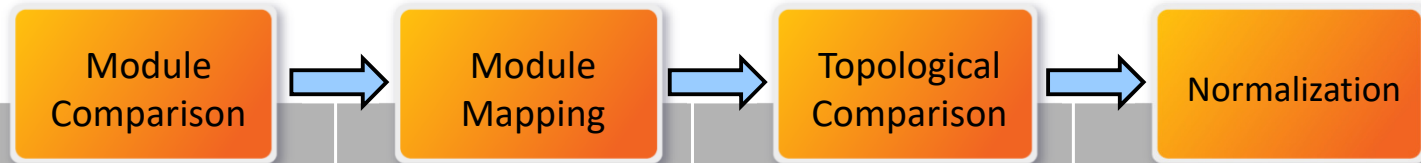
Goderis et al.

Friesen et al.

Xiang et al.

MCS = Maximum Common Subgraph GED = Graph Edit Distance

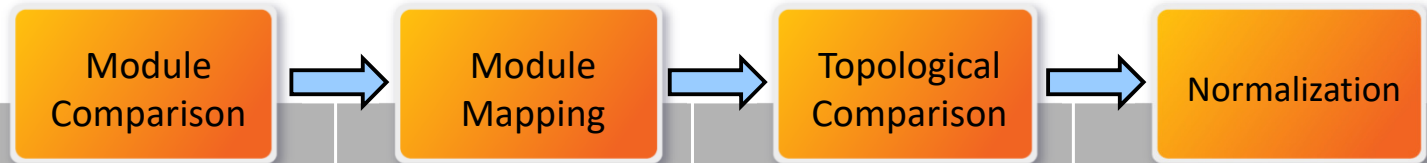
Existing Approaches



	Module Comparison	Module Mapping	Topological Comparison	Normalization
Stoyanovich et al.	single attributes	-	modules	-
Silva et al.	multiple attributes	greedy	modules	$ V $ of smaller wf
Bergmann et al.	semantic annot.	max. weight	modules & edges	$ V + E $ of query wf
	label edit dist.	max. weight	modules & edges	$ V + E $ of query wf
Santos et al.	label matching	-	modules	-
	label matching	-	MCS	$ V + E $ of larger wf
Goderis et al.	label matching	-	MCS	-
	label matching	-	MCS	'workflow sizes'
Friesen et al.	type matching	-	modules	-
	type matching	-	MCS	-
	type matching	-	graph kernels	-
Xiang et al.	label matching	-	GED	-

MCS = Maximum Common Subgraph GED = Graph Edit Distance

Existing Approaches



	Module Comparison	Module Mapping	Topological Comparison	Normalization
Stoyanovich et al.	single attributes	-	modules	-
Silva et al.	multiple attri	-	-	er wf
Bergmann et al.	semantic an	-	-	ery wf
	label edit dis	-	-	ery wf
Santos et al.	label match	-	-	-
	label match	-	-	ger wf
Goderis et al.	label match	-	-	-
	label match	-	-	es'
Friesen et al.	type matchi	-	-	-
	type matching	-	-	-
	type matching	-	graph kernels	-
Xiang et al.	label matching	-	GED	-

**What's best
At each step?
As a whole?**

MCS = Maximum Common Subgraph GED = Graph Edit Distance

Expert Curated Similarity Corpus

Reference workflow:

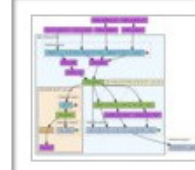


EBI_Kalign [↗](#)

Multiple sequence alignment using the Kalign tool. This workflow uses the EBI's WSKalign service (see <http://www.ebi.ac.uk/Tools/webservices/services/kalign>) to access the Kalign tool. The set of sequences to align and the molecule type (protein or nucleic acid) are the input, the other parameters for the search (see Job_params) are allowed to default.

Note: the WSKalign service used by this workflow is deprecated as of 21st September 2010 and should not be used in any new development. This service is will be retired during 2011. EBI's replacement Kalign services ([REST](#) or [SOAP](#)) should be used instead.

Are these 10 workflows similar to the reference?



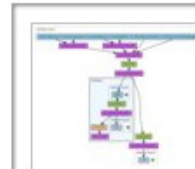
EBI_NCBI_BLAST_with_prompts [↗](#)

Run a BLAST analysis using the EBI's WSNCBIBlast service (see <http://www.ebi.ac.uk/Tools/webservices/services/ncbiblast>). This workflow wraps the EBI NCBI BLAST workflow to provide a basic



EBI_InterProScan [↗](#)

Note: the WSInterProScan web service used by this workflow is no longer available having been replaced by the EMBL-EBI's InterProScan (REST) (http://www.ebi.ac.uk/Tools/webservices/pfa/iprscan_rest)



BioQuali asynchronous workflow [↗](#)


BioQuali: Network Compatibility and products variation inference in a biological network.



Expert Curated Similarity Corpus

FlowAlike – Scientific Workflow Similarity Evaluation starling@informatik.hu-berlin.de: [dashboard](#) | [overview](#) | [help](#) | [logout](#)


Reference workflow:



EBI_Kalign [↗](#)

Multiple sequence alignment using the Kalign tool. This workflow uses the EBI's WSKalign service (see <http://www.ebi.ac.uk/Tools/webservices/services/kalign>) to access the Kalign tool. The set of sequences to align and the molecule type (protein or nucleic acid) are the input, the other parameters for the search (see tab-params) are allowed to default.

Are these 10 workflows similar to the reference?



EBI_NCBI_BLAST_with_prompts [↗](#)

Run a BLAST analysis using the EBI's WSNCBIBlast service (see <http://www.ebi.ac.uk/Tools/webservices/services/ncbiblast>). This workflow wraps the EBI NCBI BLAST workflow to provide a basic

●
●
●
●
●

- If the workflows are the same or almost the same, select **very similar**.
- If you find the two workflows to be similar, select **similar**.
- If the workflow doesn't offer similar but highly related functionality, e.g., it includes or is included in the reference workflow, select **related**.
- If you are unsure whether the workflows are similar or not, select **unsure**.
- If you find the workflows to be rather NOT similar, select **dissimilar**.


⋮

Expert Curated Similarity Corpus

FlowAlike – Scientific Workflow Similarity Evaluation starling@informatik.hu-berlin.de: [dashboard](#) | [overview](#) | [help](#) | [logout](#)


Please choose a reference workflow to rate similar workflows against

Stage 1



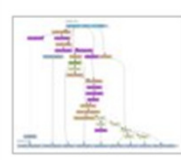
1. HUMAN Microarray CEL file to candidate pathways [↗](#)

» Review



4. Extract proteins [↗](#)

» Rate




7. Workflow for Protein Sequence Analysis [↗](#)

» Rate


Stage 2



2. Some cat and acc



5. K ana



8. S Blas Par

Legend:

- very similar
- similar
- related
- unsure
- dissimilar

- 24 query workflows
- Each with 10 other workflows to compare to it
- + Extended comparison lists for specific algorithms' results for 8 query workflows
- **15 experts (7 institutes)** provided
- > **2400 ratings**
 - classifying each pair of workflows
 - ranking workflow lists by similarity

Results

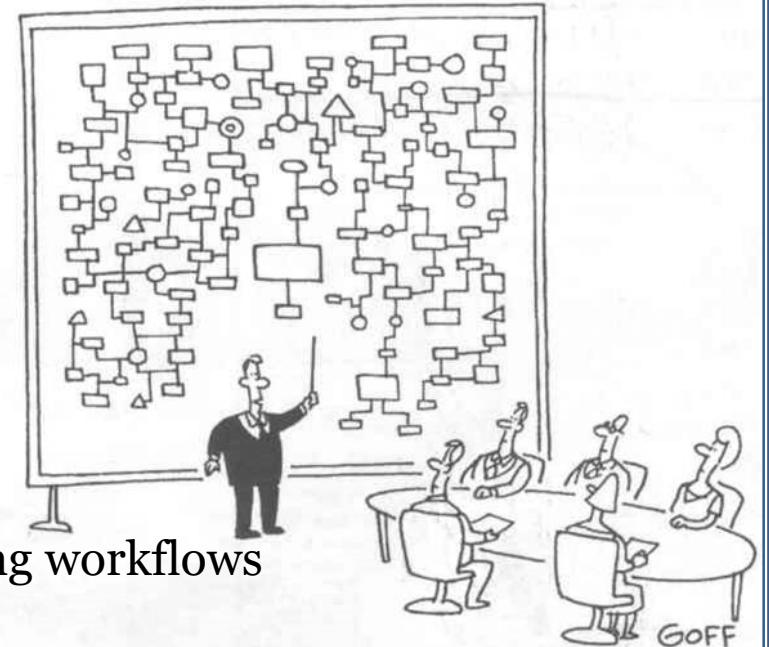
- ▶ **Experts agreed** on the similarity of workflow pairs
- ▶ **Annotation-based** approaches
 - Provide best results
 - But **only a few** well-annotated workflows
- ▶ **Structural approaches**
 - Outperform annotation-based
 - Galaxy & VisTrails
 - Graph edit distance is too expensive
 - Module set provides good results
 - **Room for solutions in between**
 - LayerDecomposition [**eScience 2014**]
with **J. Starlinger**, U. Leser, S. Davidson, S. Khanna
 - Usable in real environments (myExperiment)
[**Future Generation Computer System 2016**]

How to improve reuse?

Help finding
similar
workflows



Make
workflow
structures
less complex!



Plumbing workflows

DistillFlow

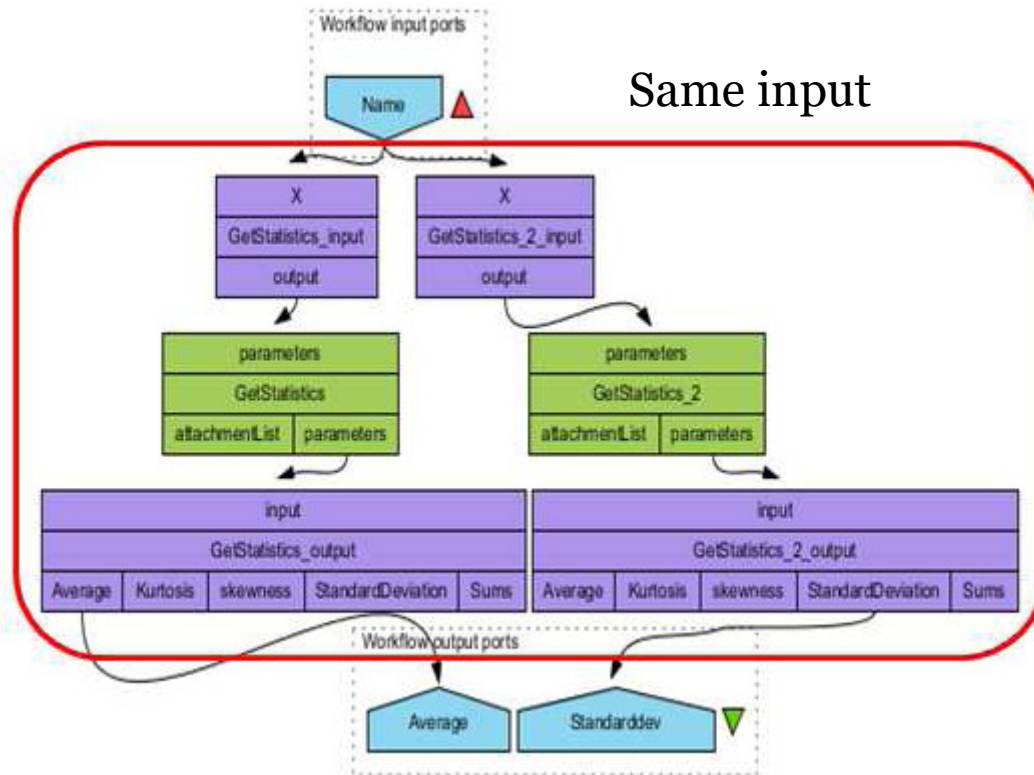
- ▶ Distilling workflow structures: Removing redundancy
- ▶ Collaboration with Taverna & BioVel
- ▶ BioVel (FP7)



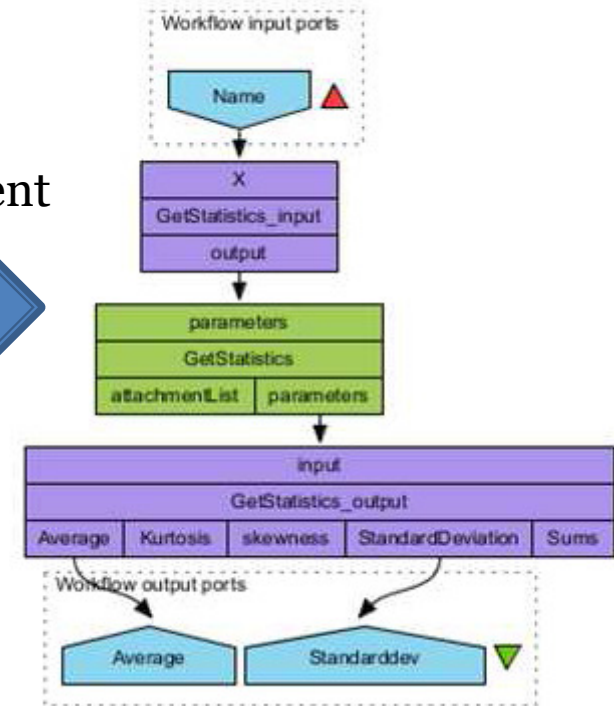
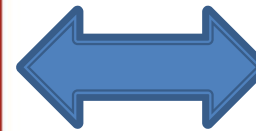
- Virtual laboratory: Libraries of workflows for research on biodiversity
- Consortium of 15 partners (9 countries)

- Improving reuse in BioVel
- More generally: improving reuse in Taverna

Use case 1



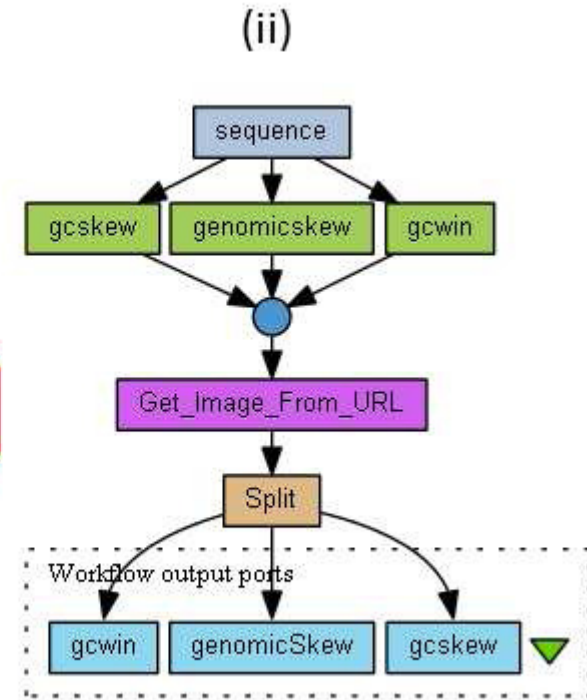
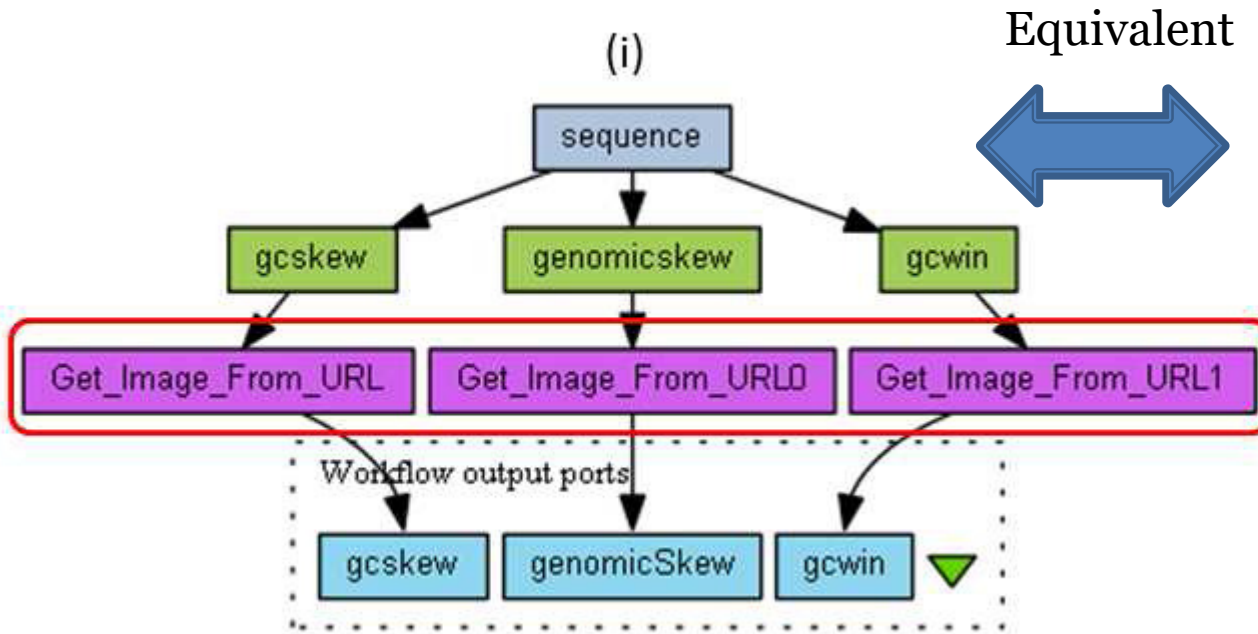
Equivalent



3 processors duplicated!
→ Pure redundancy

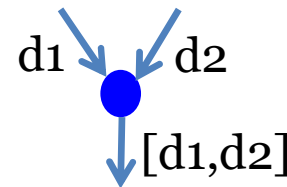
No redundancy

Use case 2

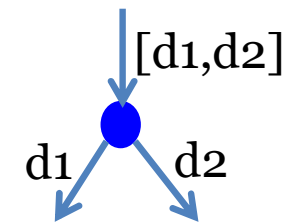


Workflow (ii) uses

merge



split



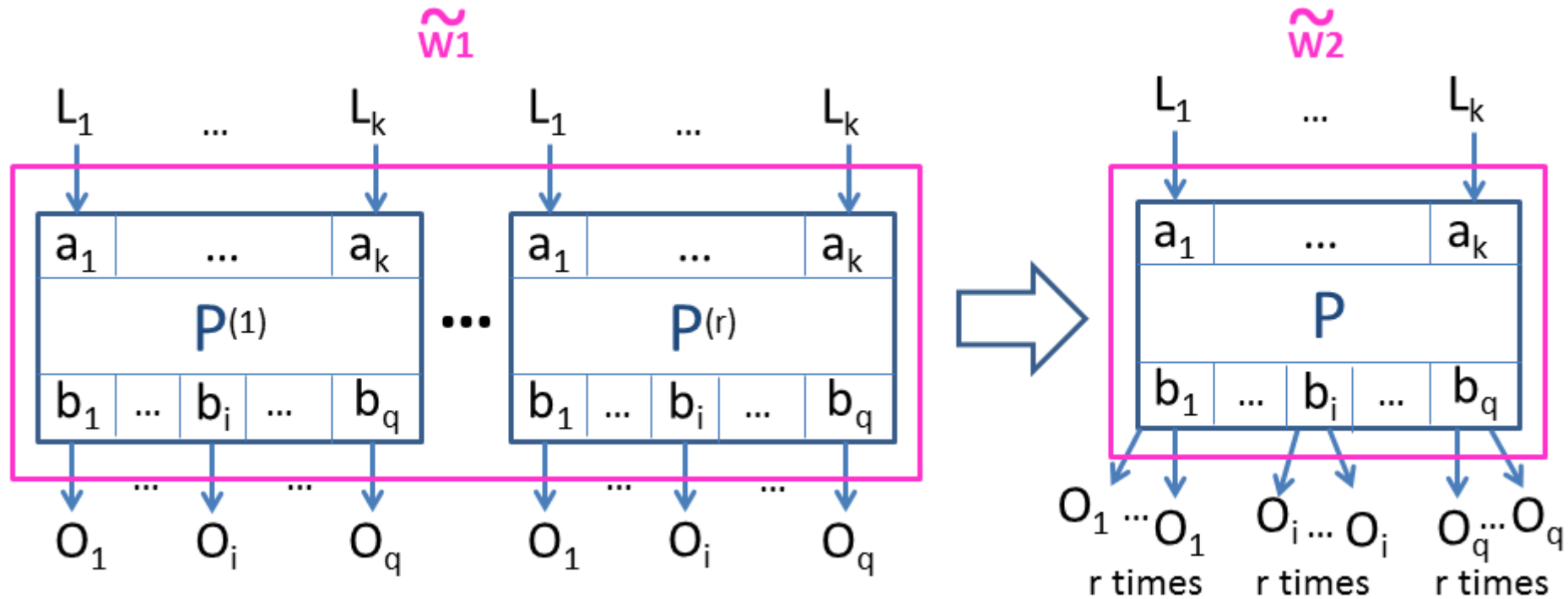
Rewriting workflows

- ▶ Exploring the **implicit iteration** feature of Taverna
 - List of items with merge/split instead of single items with duplication
- ▶ Assumptions before merging several copies of a processor
 - Only copies with the **exact same code**
 - Only copies that **do not depend on each other**
 - Only **deterministic** processors (same input → same output)

→ 2 **anti-patterns** and the corresponding rewriting

Anti-pattern (A)

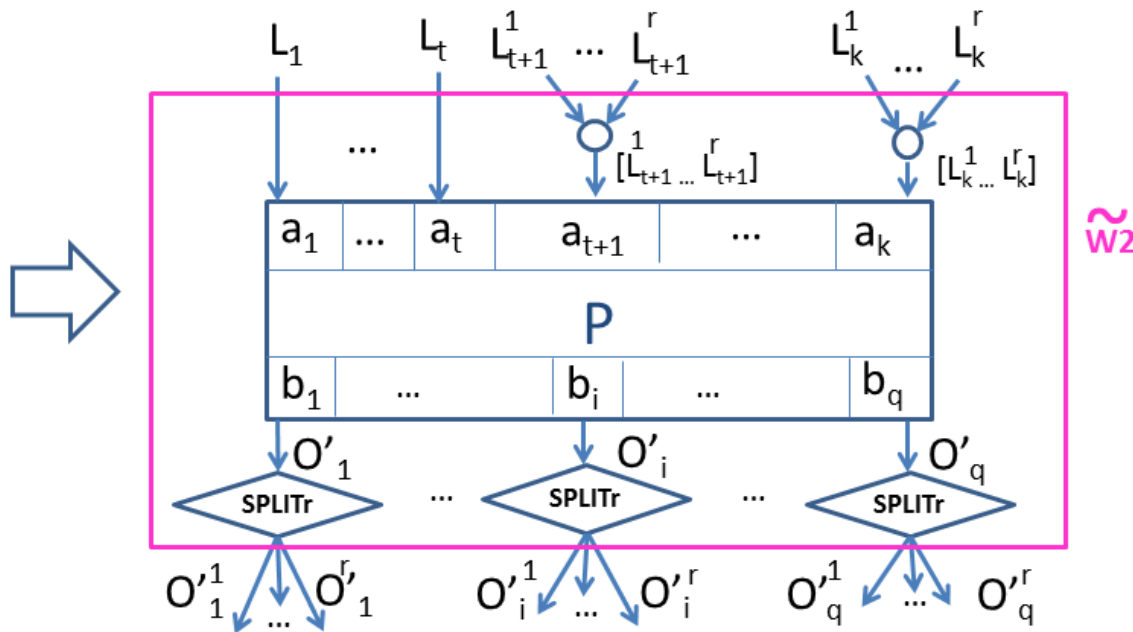
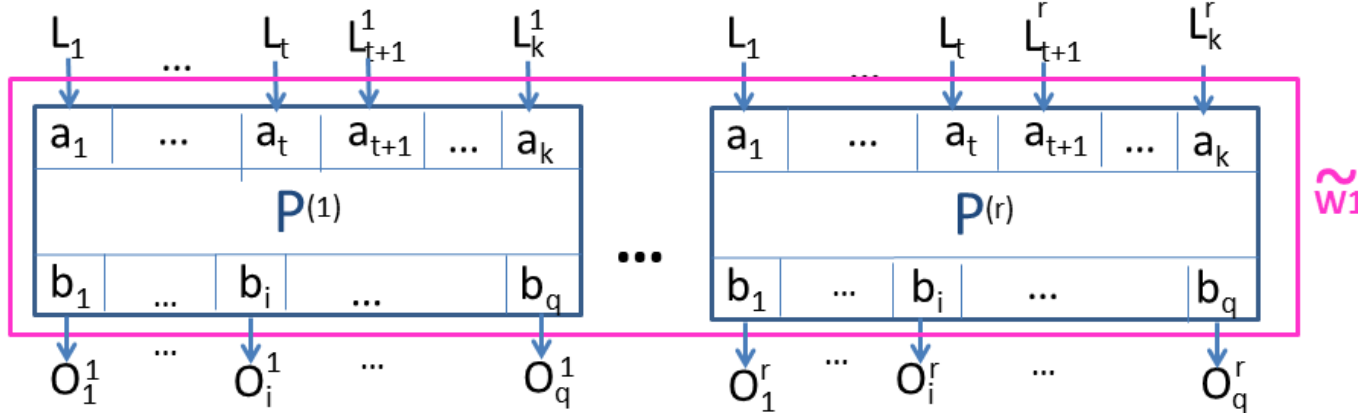
Corresponds to use case 1



L_i can be one single value or a list of values

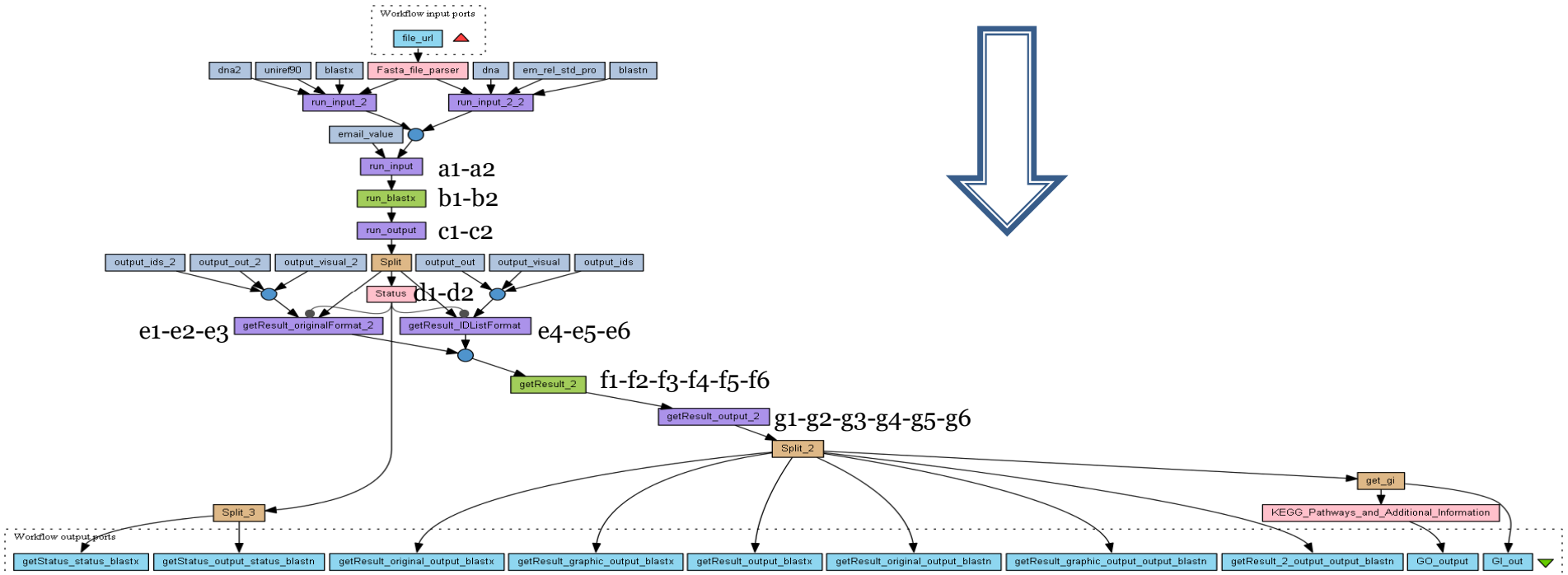
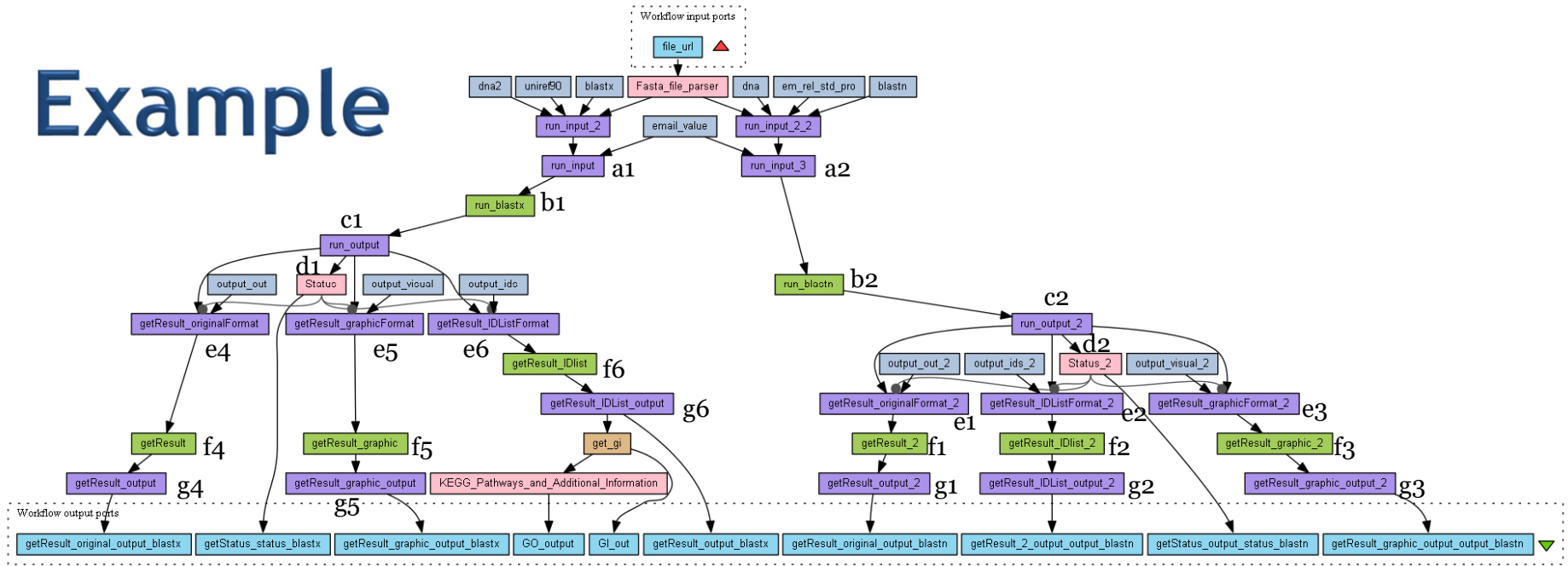
Anti-pattern (B)

Corresponds to use case 2



Processor P applies **cross product** to values on ports a_1 to a_t and **dot product** to values on ports a_{t+1} to a_k

Example



This Tutorial

- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - workflows and reproducibility
 - [Latest results on workflows](#)
 - Or How CS research may have direct impact on LS
 - Improving reuse
 - [Managing Provenance](#)
 - Comparing workflows executions

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

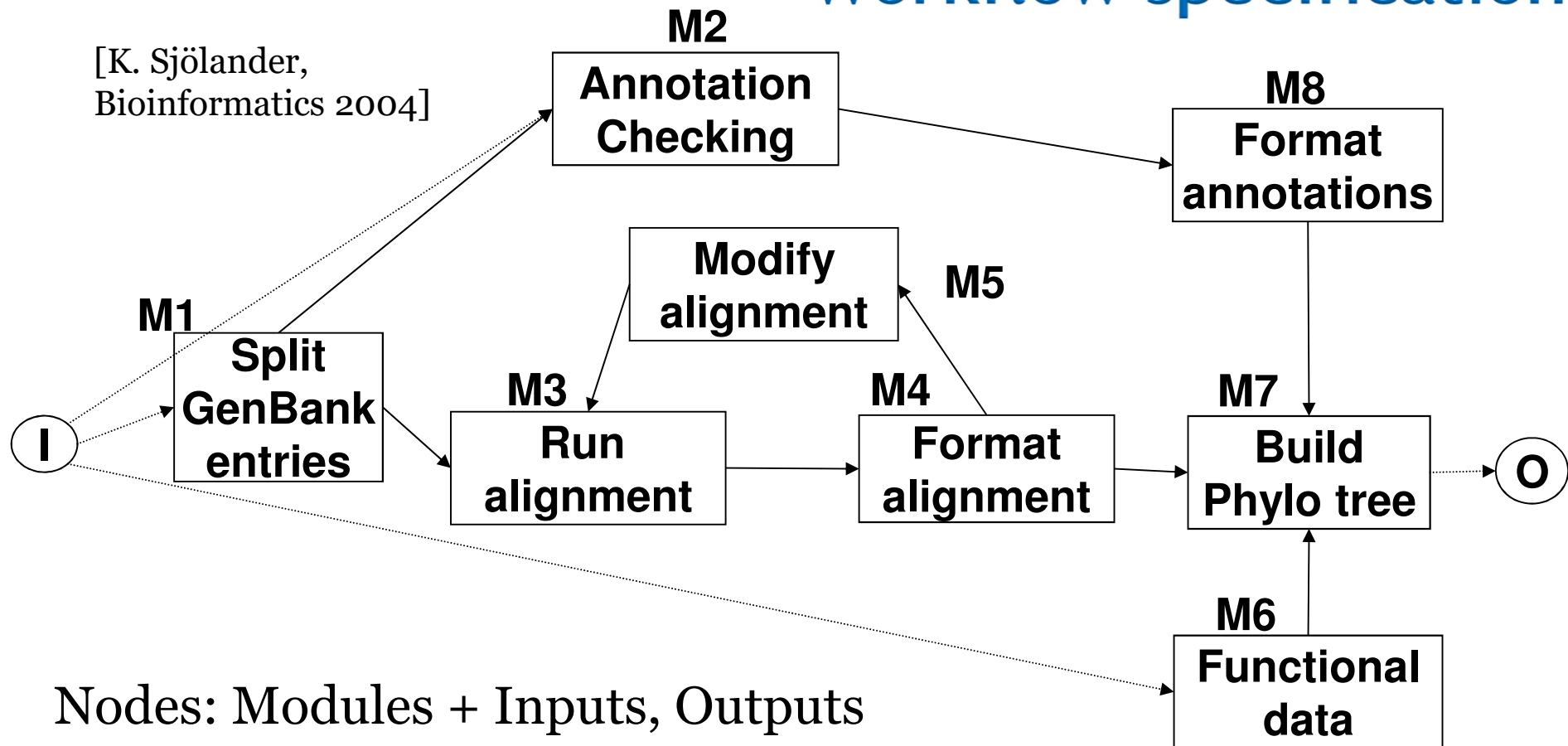
- ▶ **Part III – Conclusions**

Provenance in scientific workflows

- ▶ Provenance is highly important for users to **interpret** any scientific result
- ▶ Workflow systems are now equipped of *Provenance Modules* capturing the exact set of data used and consumed by the execution of each workflow step
- ▶ **Standards** to represent provenance information are now defined (W3C)
- ▶ One of the major challenge lies in dealing with the **huge amounts of information**
 - Example of solution with ZOOM*userviews which use the composition to hide (part of) the data

Workflows are graphs

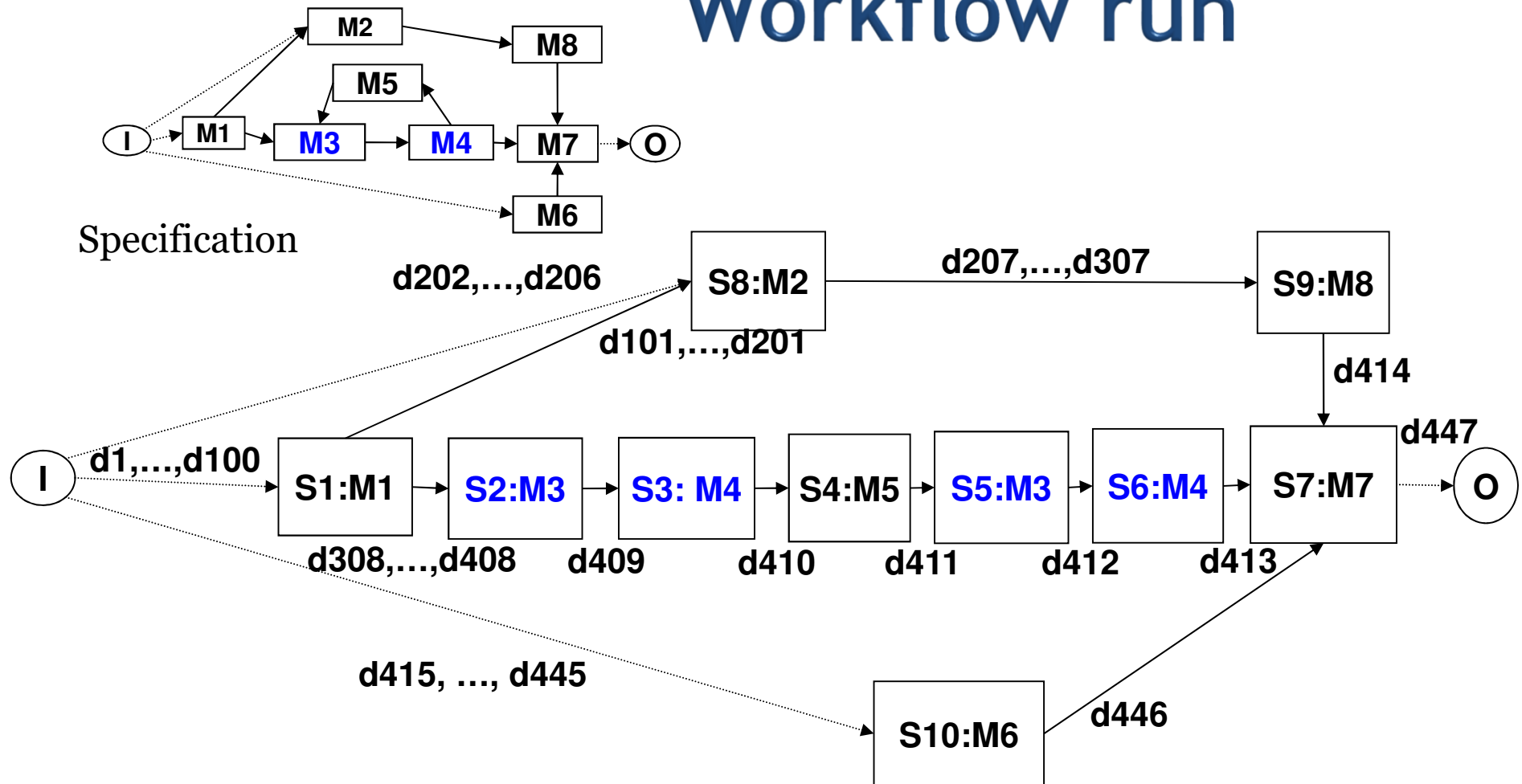
Workflow specification



Nodes: Modules + Inputs, Outputs

Edges: Possible dataflow

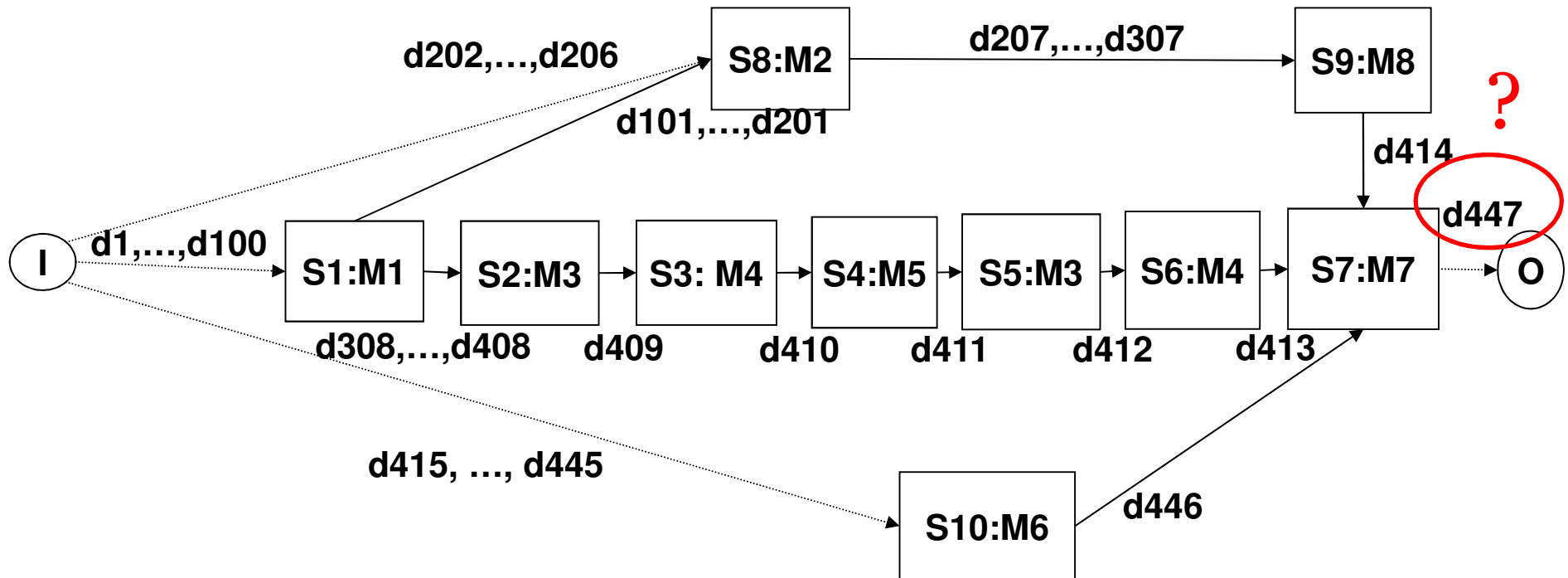
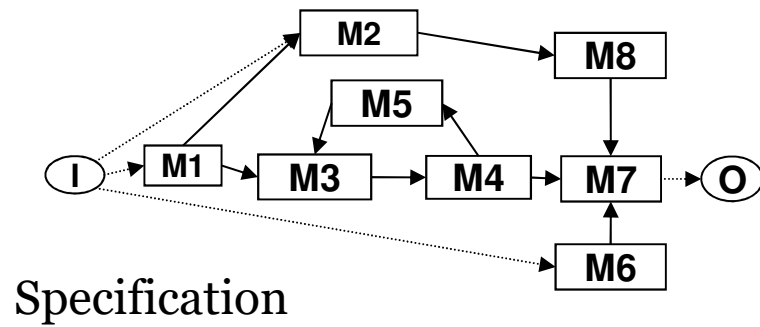
Workflow run



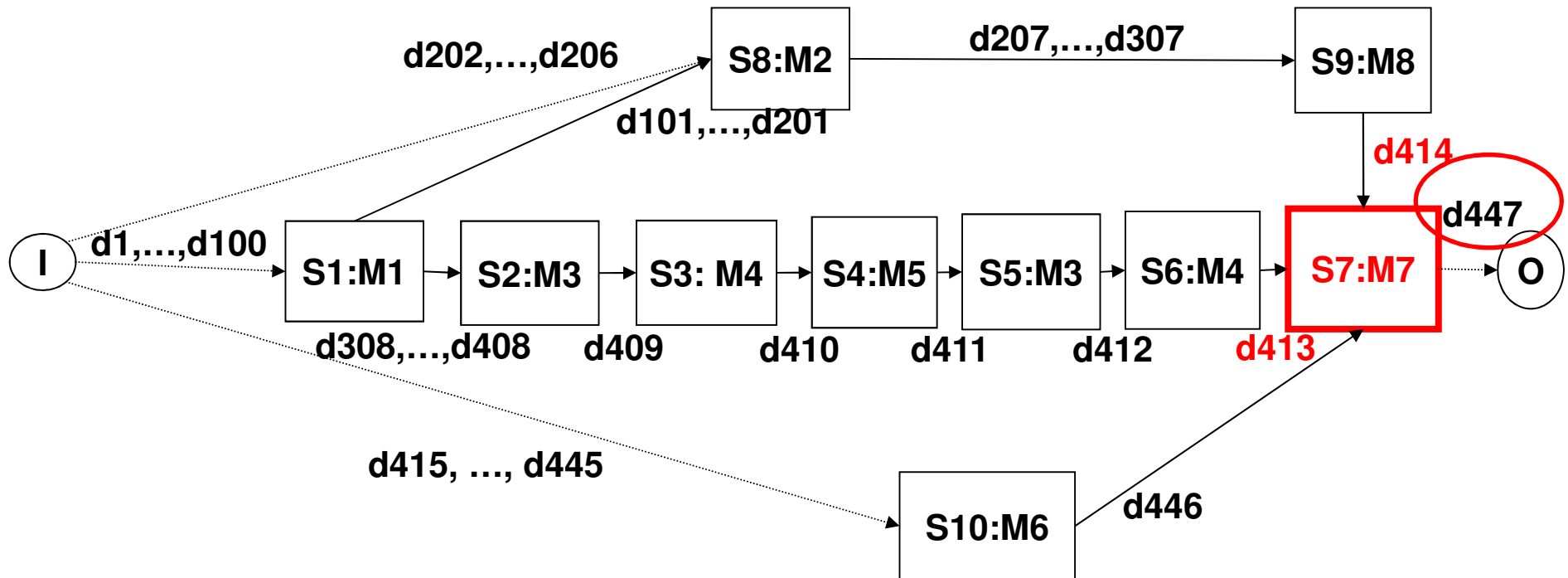
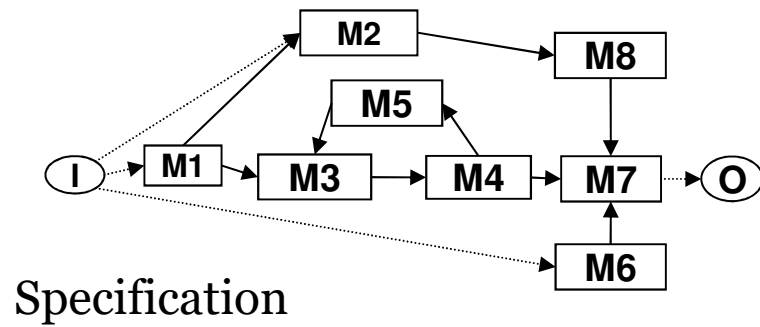
Nodes: Steps (executions of modules)

Edges: Actual dataflow (labelled with data object ids)

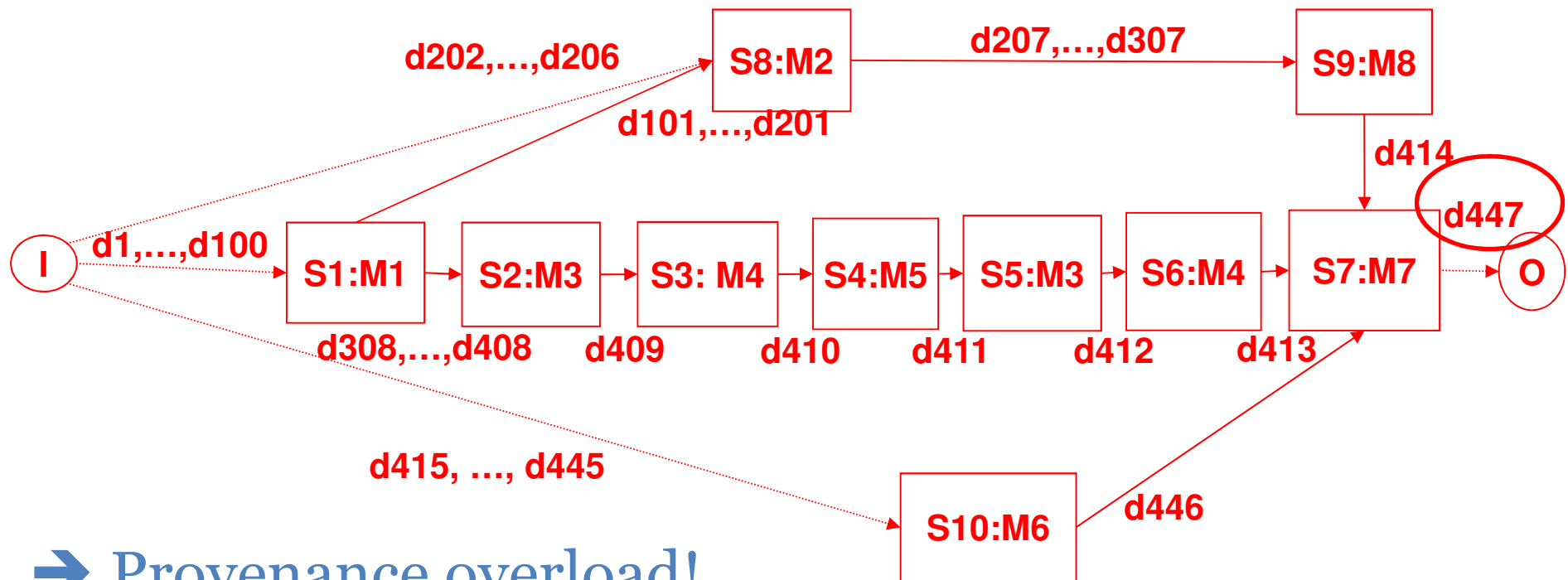
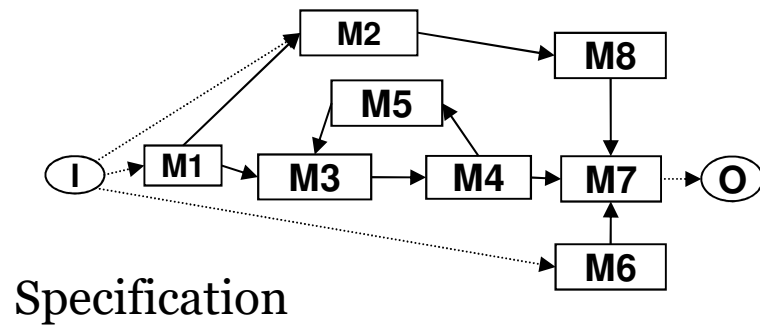
Workflow run: Provenance of d447? (tree generated)



Workflow run: Provenance of d447 (immediate)



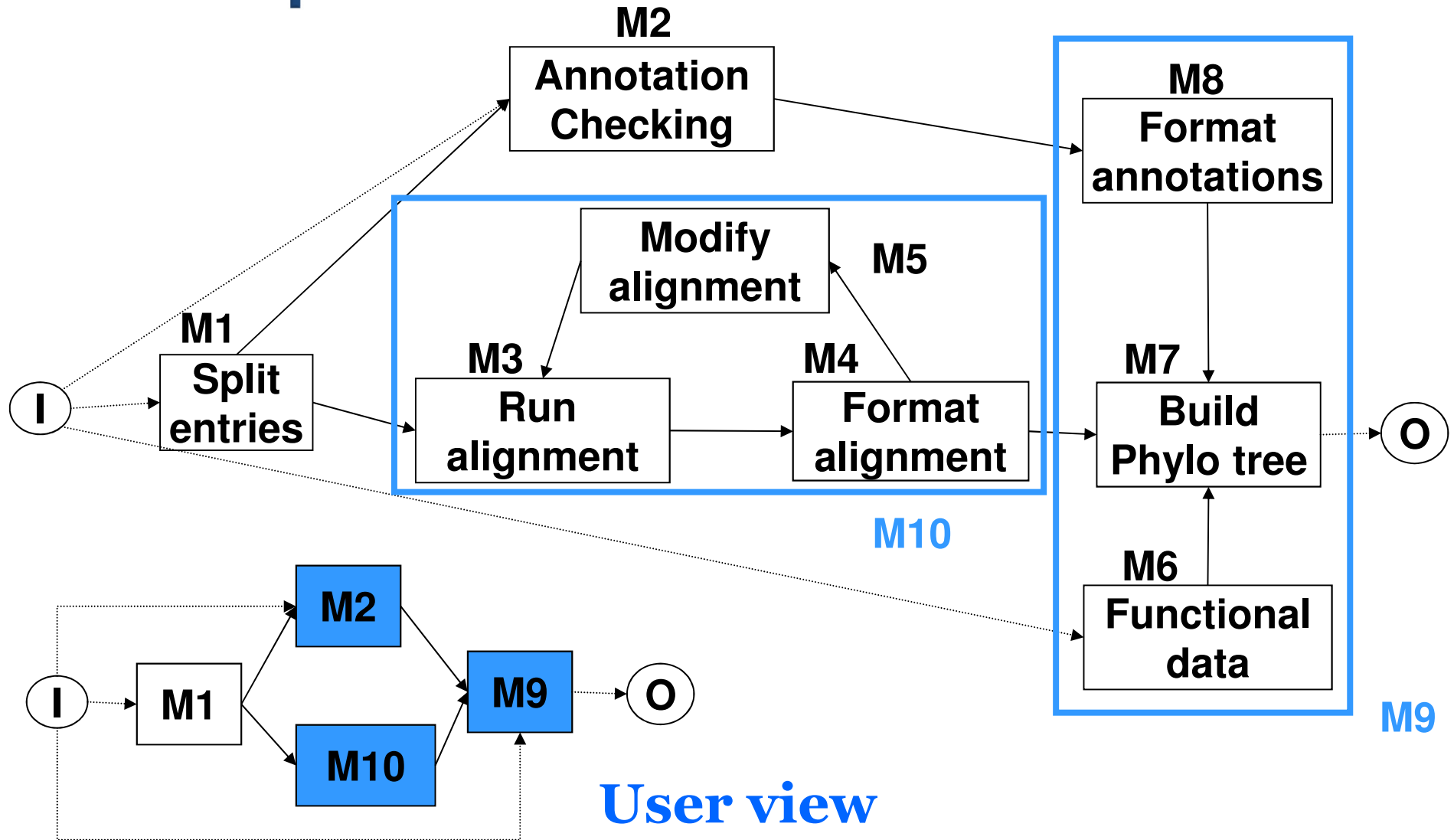
Workflow run: Provenance of d447 (deep)



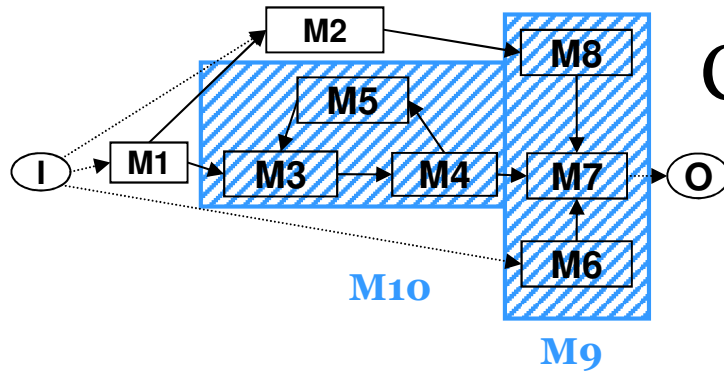
➔ Provenance overload!

➔ Need to focus on relevant information

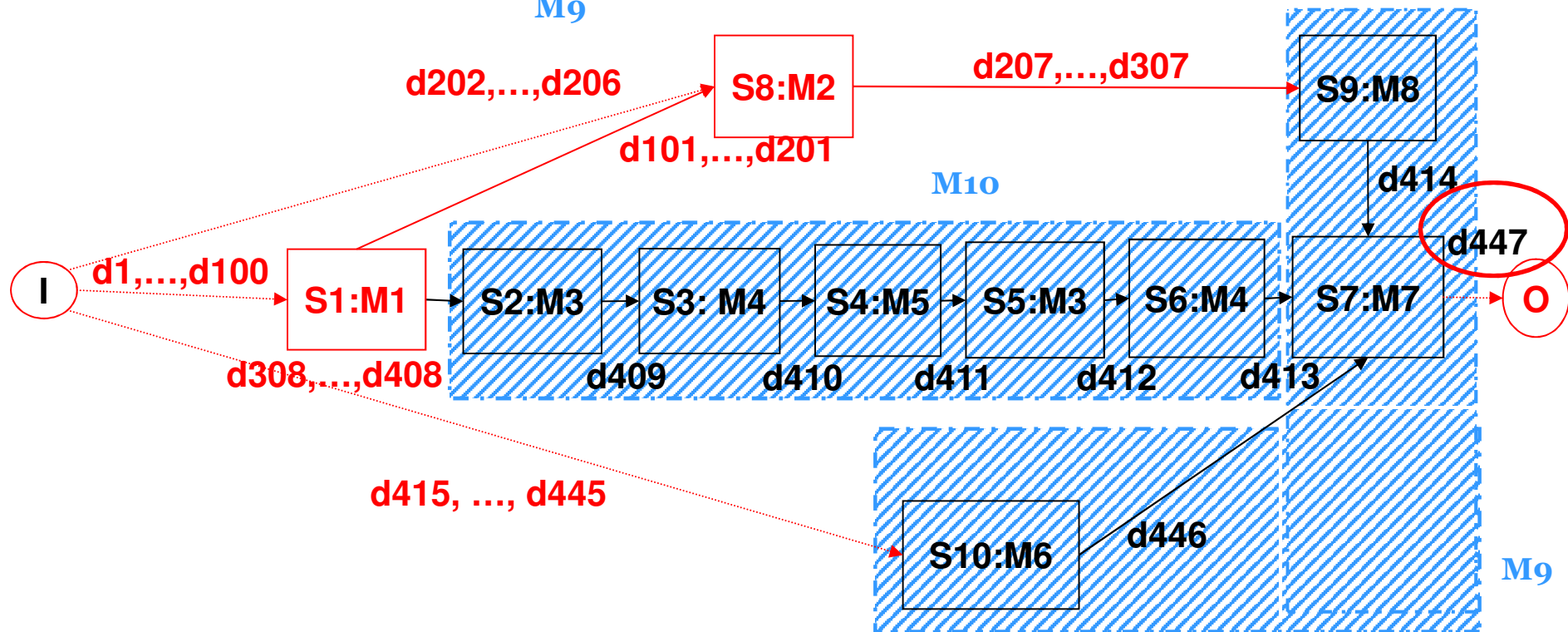
Composite modules



Composite modules



Composition simplifies provenance!

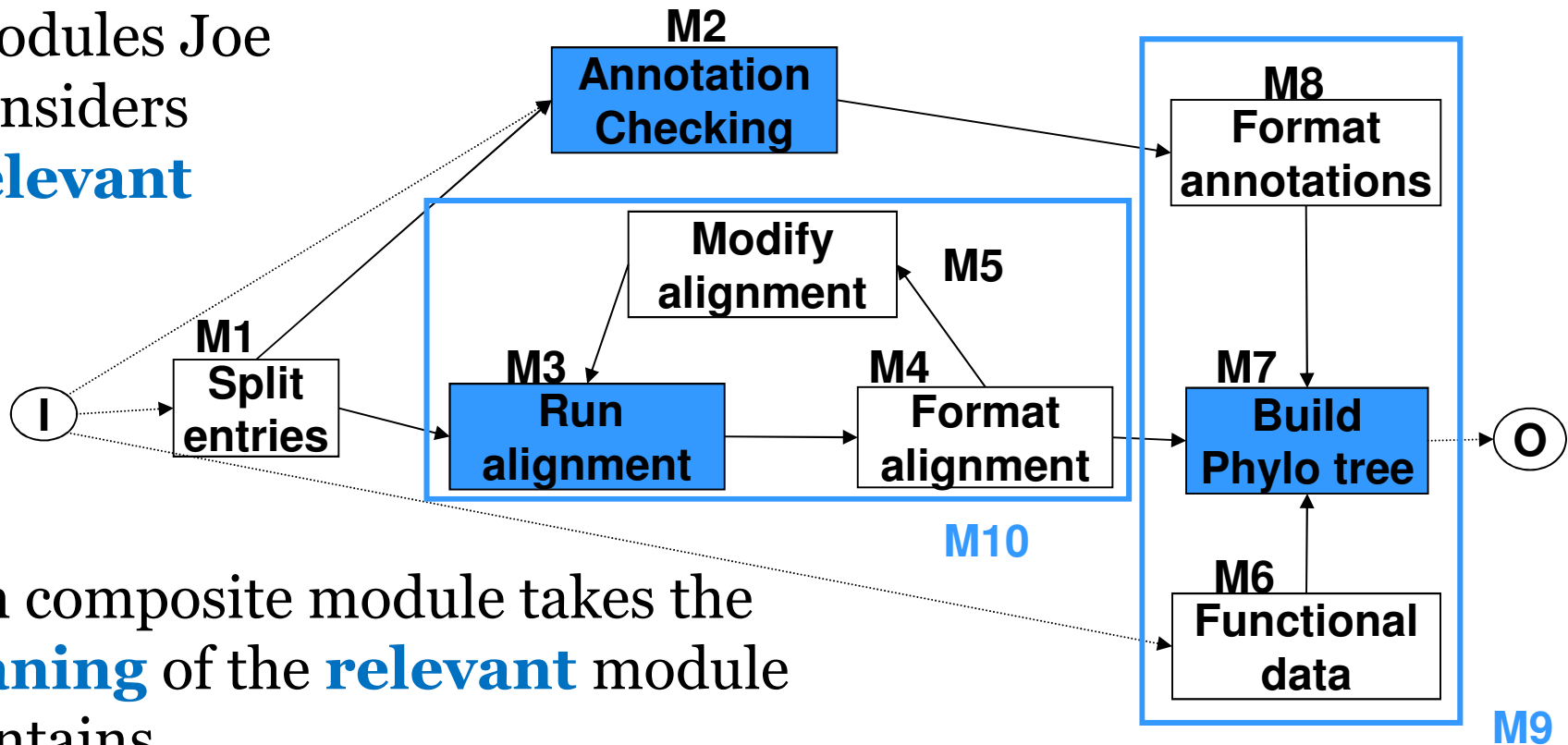


Designing composite modules

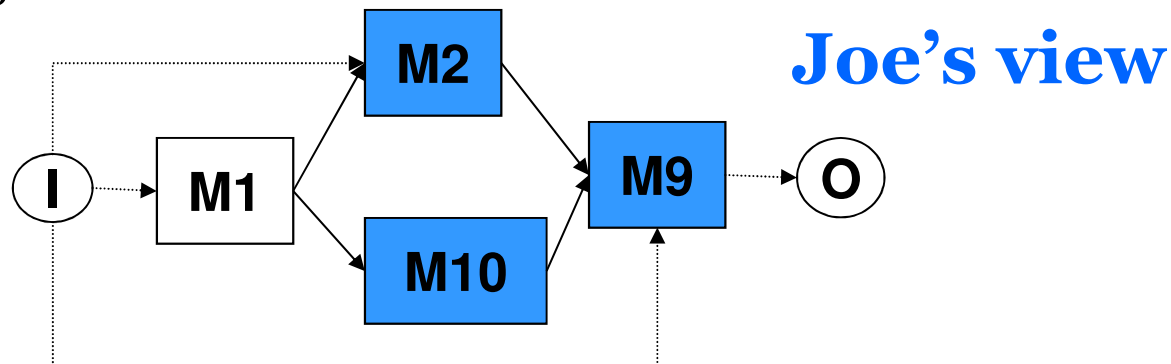
- ▶ Composite modules are typically defined **by the workflow designer** to
 - Enable **reuse** between workflows
 - **Simplify** the view of the workflow according to what modules the **designer thinks are relevant** in the workflow
- ▶ However, users may have **different interests**, i.e. have different relevant modules
- ➔ **Several user views** of a given workflow should thus be considered, constructed according to each user's interest

Relevant user view

Modules Joe considers **relevant**

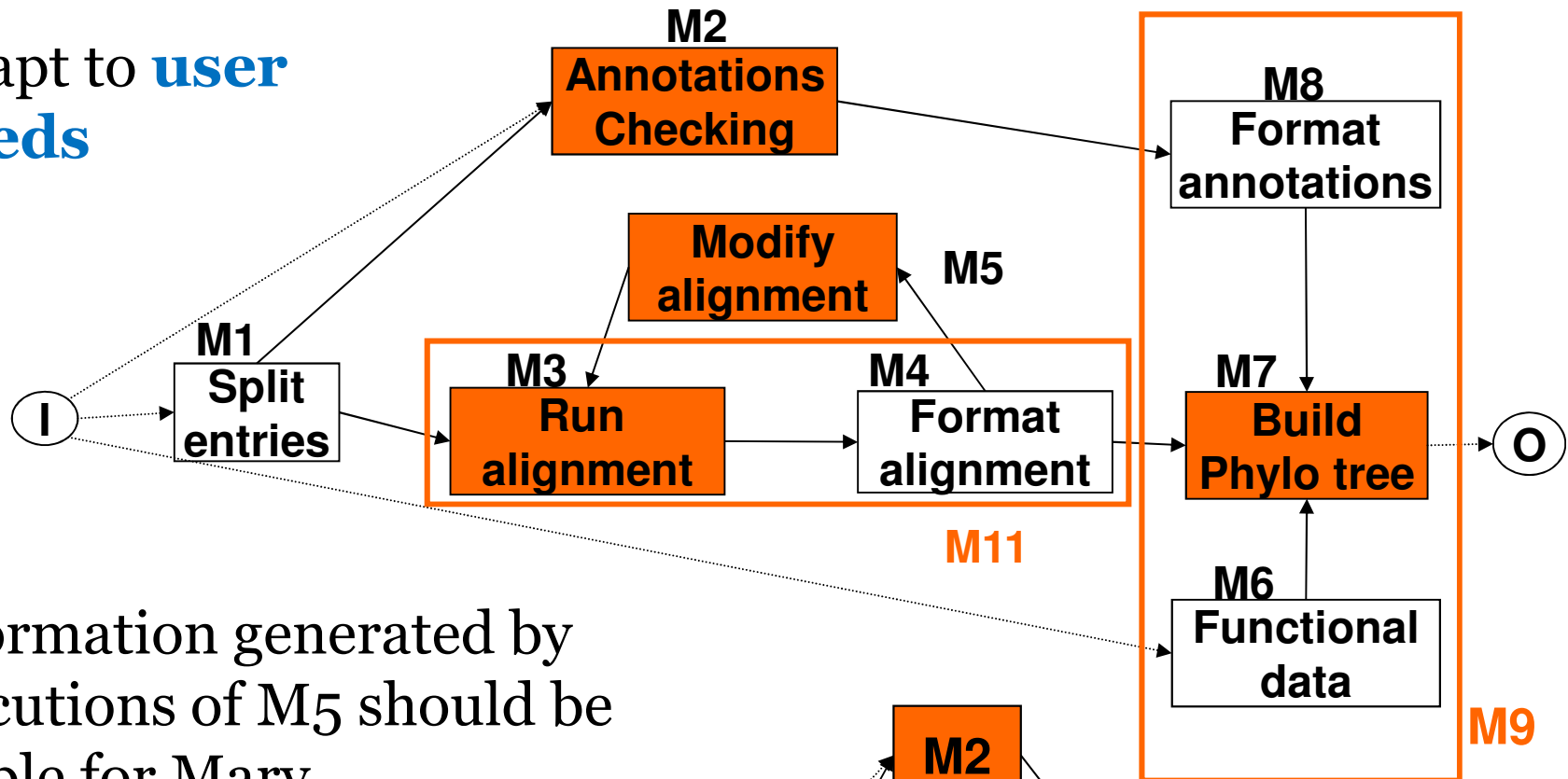


Each composite module takes the **meaning** of the **relevant** module it contains



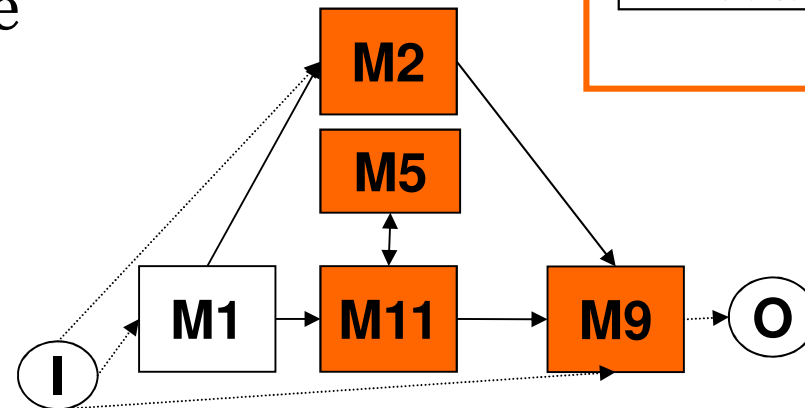
User views may differ

Adapt to **user needs**



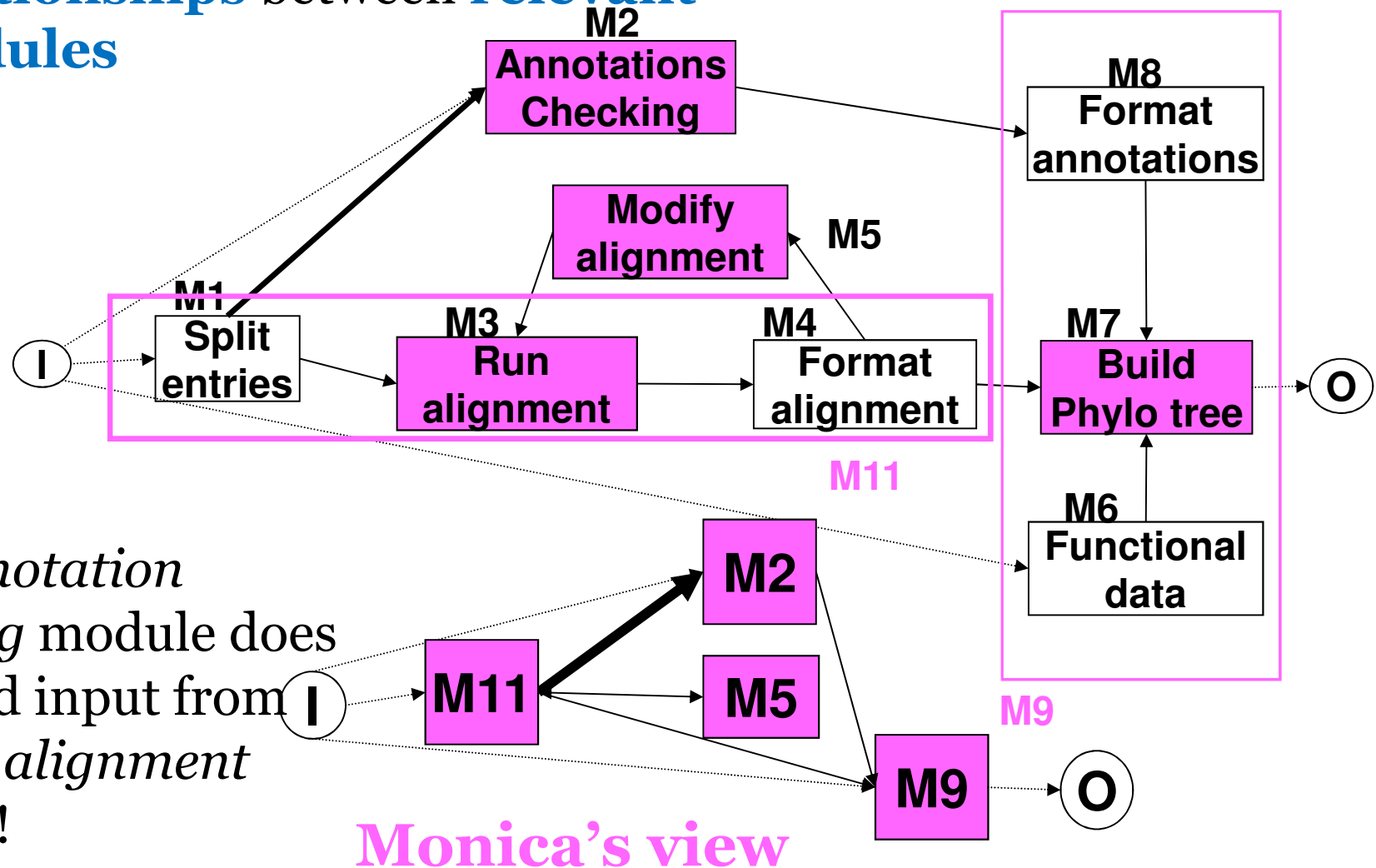
Information generated by executions of M5 should be visible for Mary

Mary's view



Grouping may be error-prone!

Grouping should **preserve the relationships** between **relevant modules**



The *annotation checking* module does not need input from the *run alignment* module!

Monica's view

ZOOM*UserViews

▶ Goals

- Help user **construct relevant user views**
 - Preserving the **relationships** between **relevant modules**
- Exploit **user views** to **reduce the provenance information** returned as answer to a query

▶ Contributions

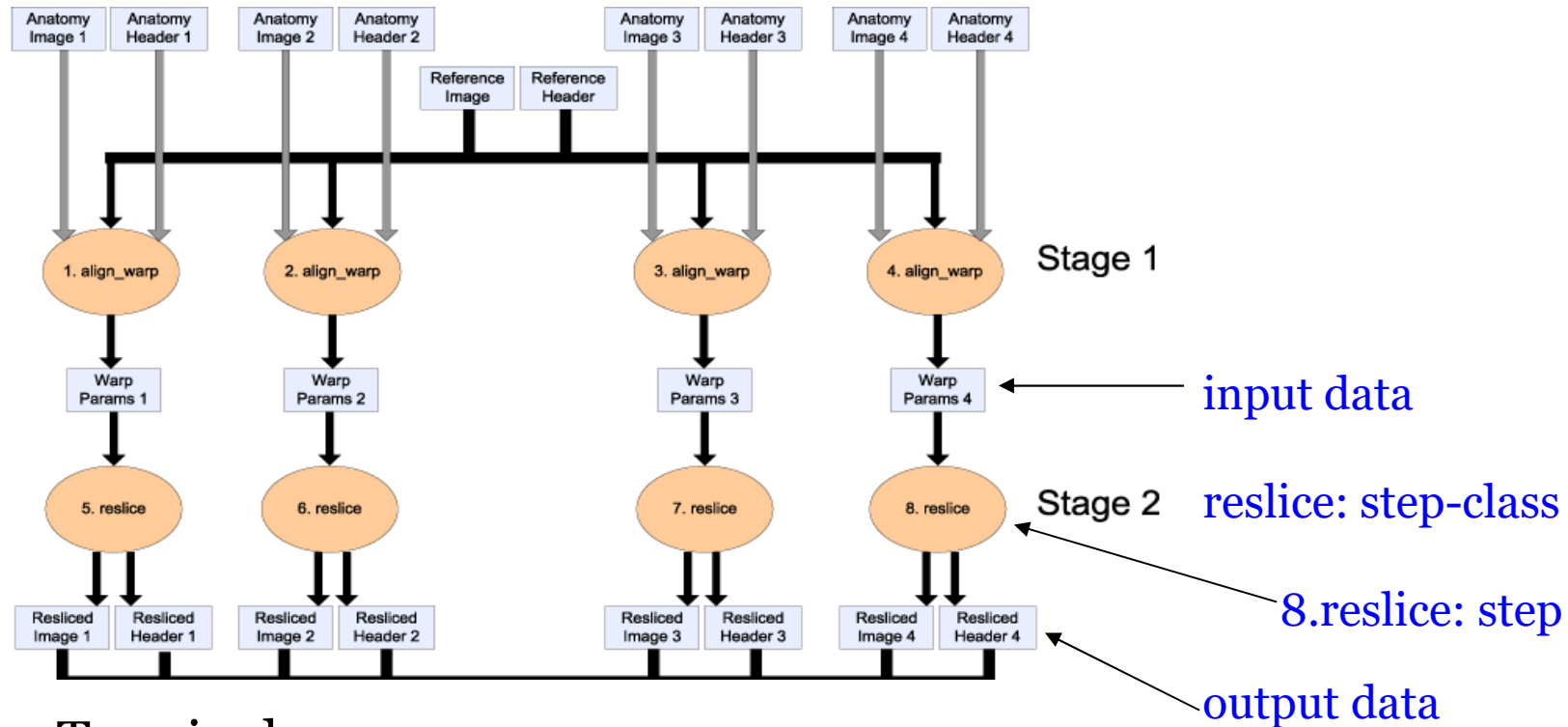
- **Model** for provenance and user views in scientific workflows
- **Algorithm (polynomial)** for generating **relevant** user views according to the user's interests (minimal)
- **Provenance Reasoning system:** Querying provenance **through user views**



Provenance challenge

- First Provenance Challenge (**twiki.ipaw.info**)
 - By S. Miles, M. Wilde, I. Foster and **L. Moreau**, at Washington DC, Sept. 2006
- **Aims:** Understanding the **capabilities** of provenance-related systems (17)
- The **challenge process**
 - **Workflow example (spec + run)** provided
 - **List of provenance queries** to be answered

Workflow Representation



Terminology

- Nodes are **step-classes** (static)
- Edges capture the **flow of data** between step-classes
- An **execution** of a workflow generates a partial order of steps (dynamic)
 - Instances of step classes
- Each step has **input** and **output** data

This Tutorial

- ▶ **Part I – Data Integration workflows**
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - workflows and reproducibility
 - **Latest results on workflows**
- Or How CS research may have direct impact on LS
 - Improving reuse
 - Managing Provenance
 - Comparing workflows executions**

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

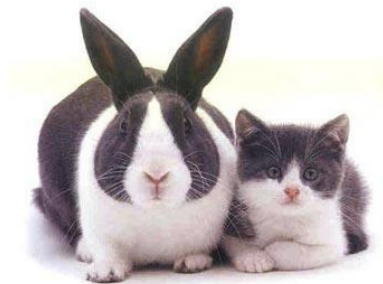
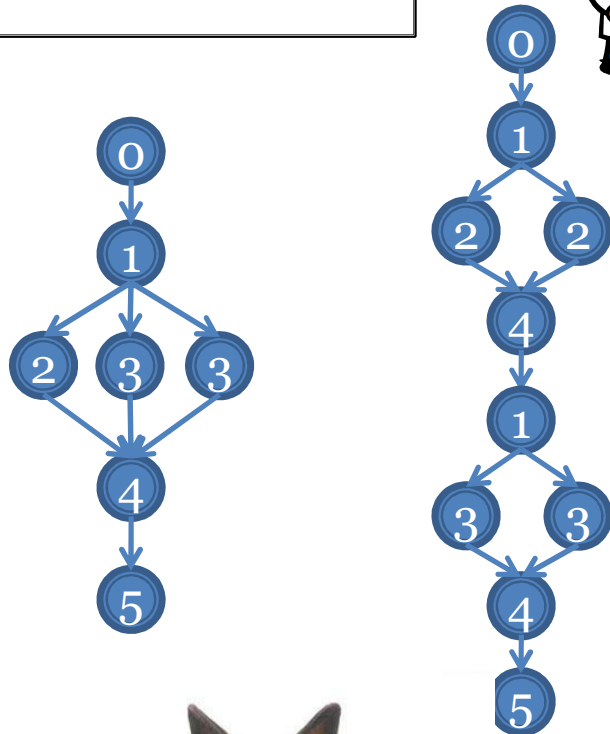
- ▶ **Part III – Conclusions**

Workflow runs Difference Problem

What's the difference between these two runs of the same workflow?



Our problem is more than a "spot the difference" puzzle!

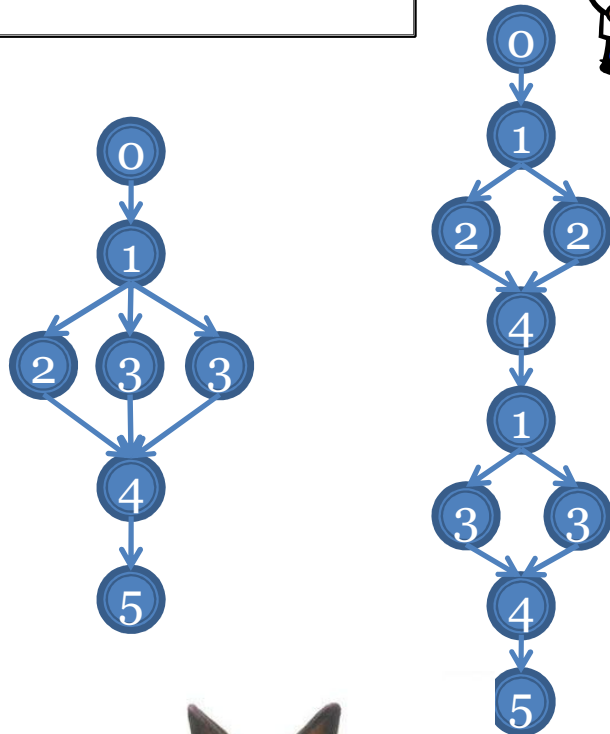


Workflow runs Difference Problem

What's the difference between these two runs of the same workflow?



Our problem is more than a "spot the difference" puzzle!



Mapping different objects in two figures is trivial.

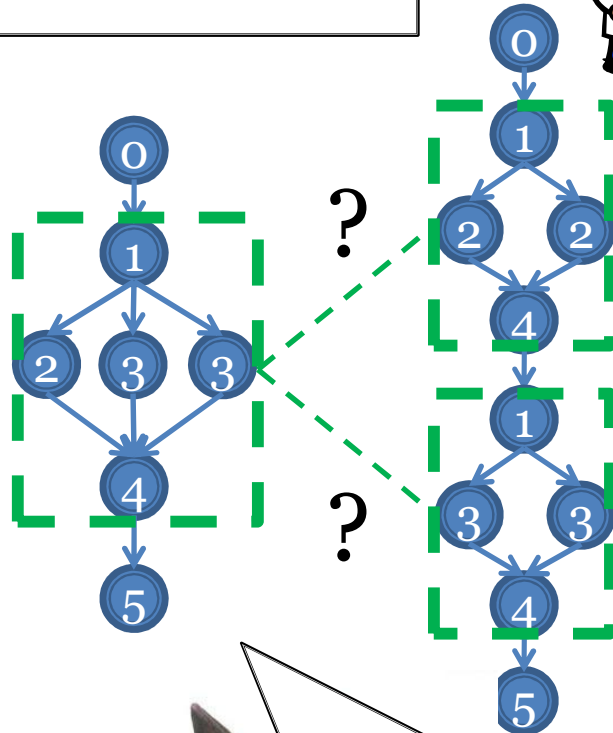


Workflow runs Difference Problem

What's the difference between these two runs of the same workflow?



Our problem is more than a "spot the difference" puzzle!



Mapping different fork or loop copies in two runs is **nontrivial!**

Mapping different objects in two figures is trivial.

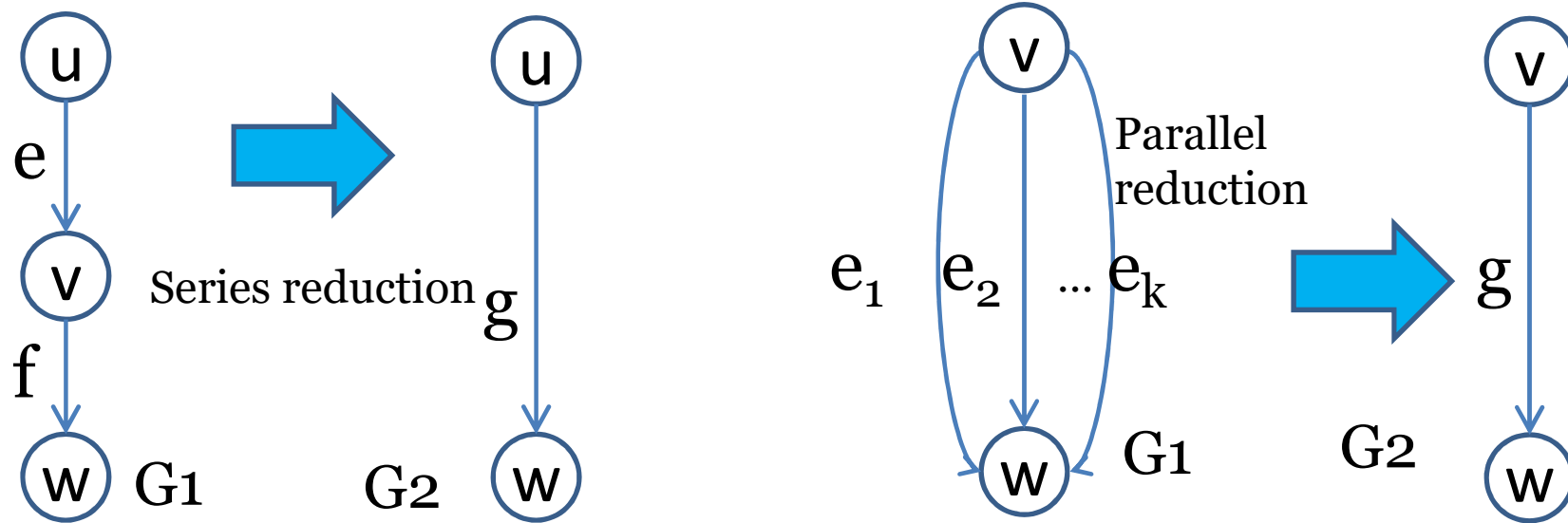
Workflow runs Difference Problem

The problem of differencing runs
is NP-hard on DAGs
while polynomial time algorithms
can be designed for
Series-Parallel (SP) structures
→ Some approaches have
considered such restrictions on
workflow graph structures

Definition of SP-graphs

G is SP iff $\text{MaxRed}(G) = \text{BSP}$

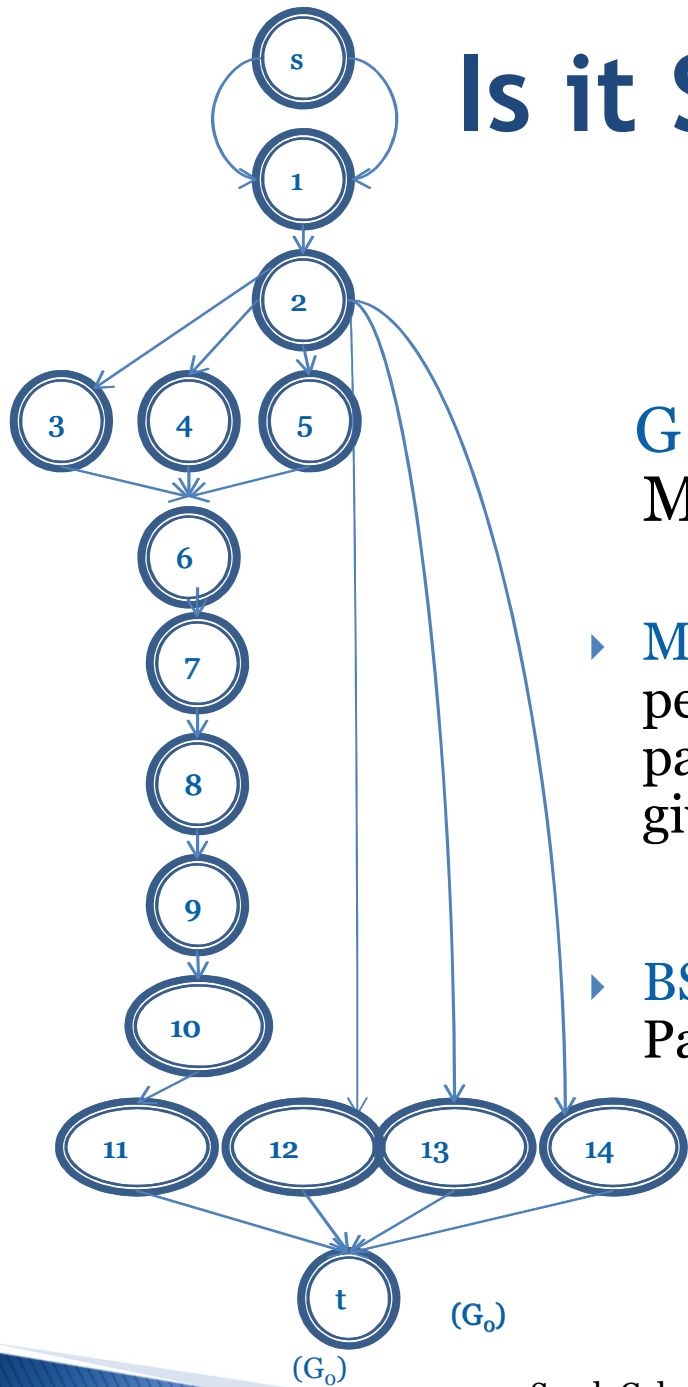
- ▶ **MaxRed(G)**: iteratively performs series and parallel reductions on a given graph G



- ▶ **BSP**: Basic Series-Parallel



Is it Series-Parallel?



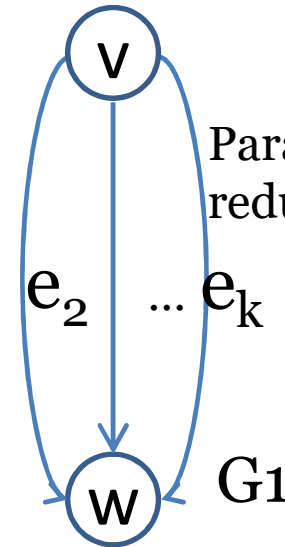
G is SP iff
 $\text{MaxRed}(G) = \text{BSP}$

▶ **MaxRed(G)**: iteratively performs series and parallel reductions on a given graph G

▶ **BSP**: Basic Series-Parallel

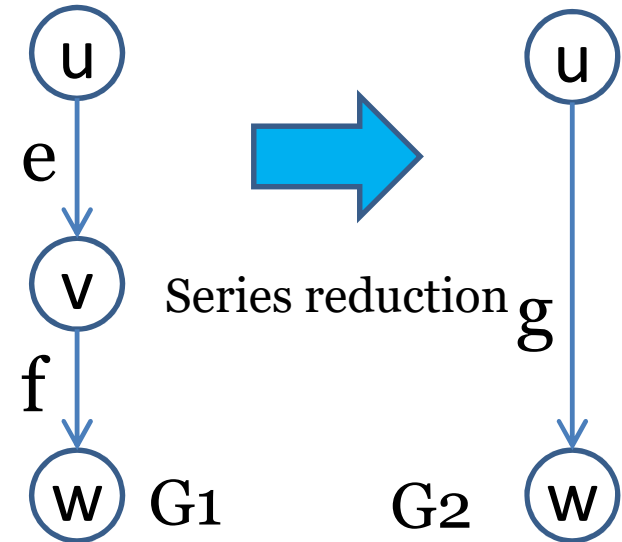
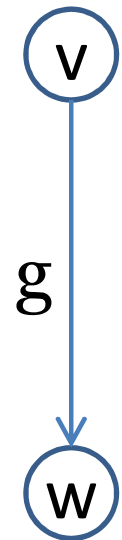


e_1



Parallel reduction

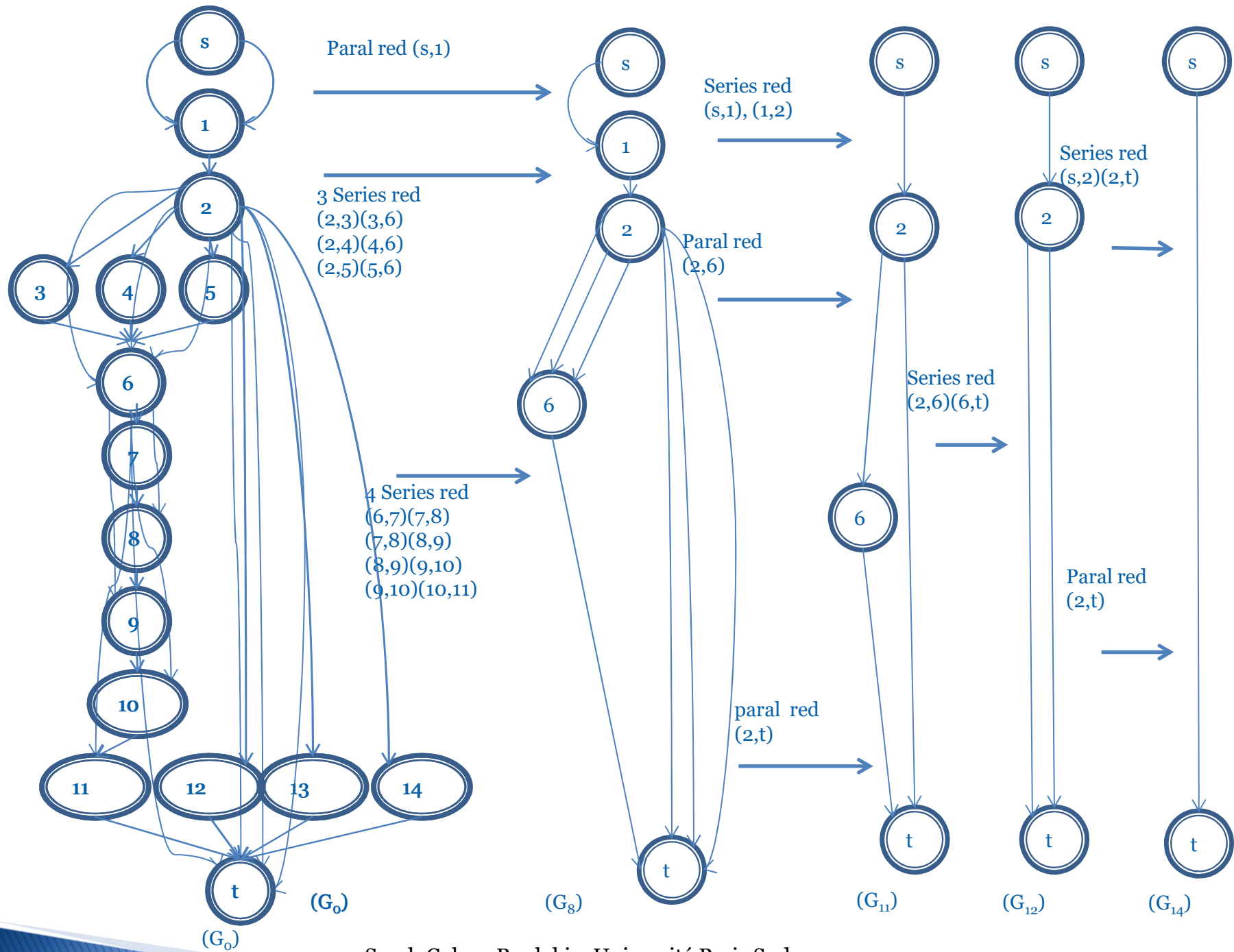
G_2



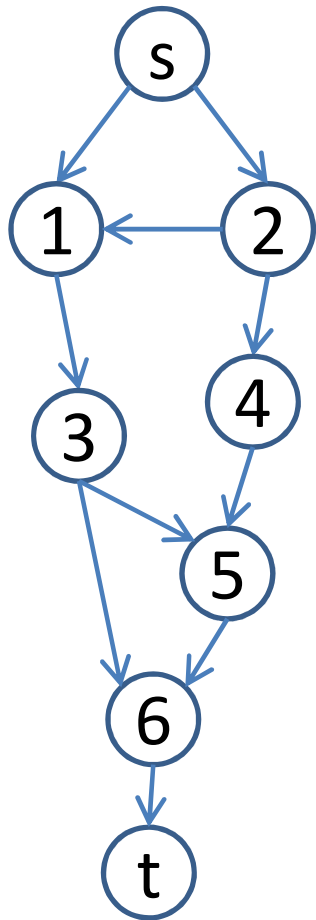
Series reduction g

G_2





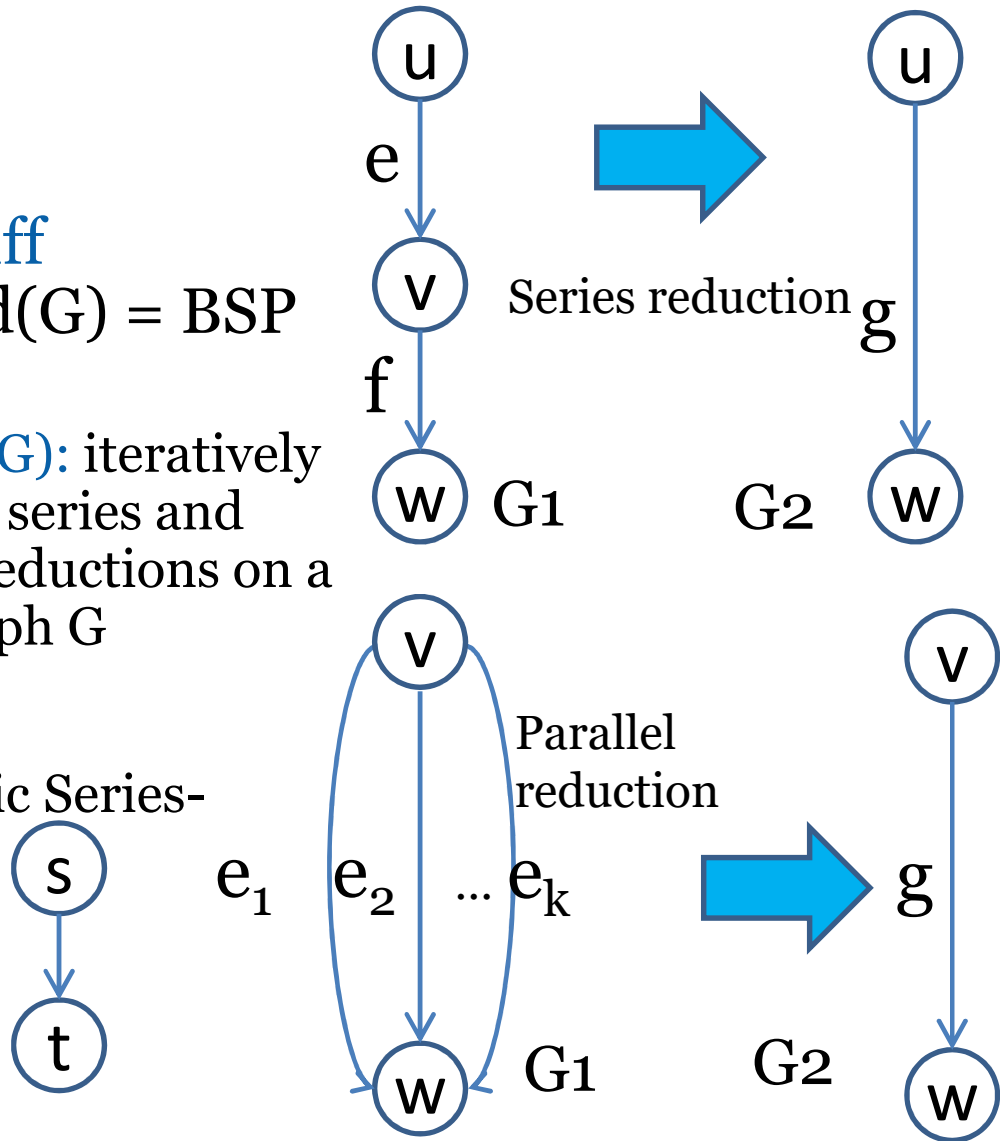
Is it Series-Parallel?



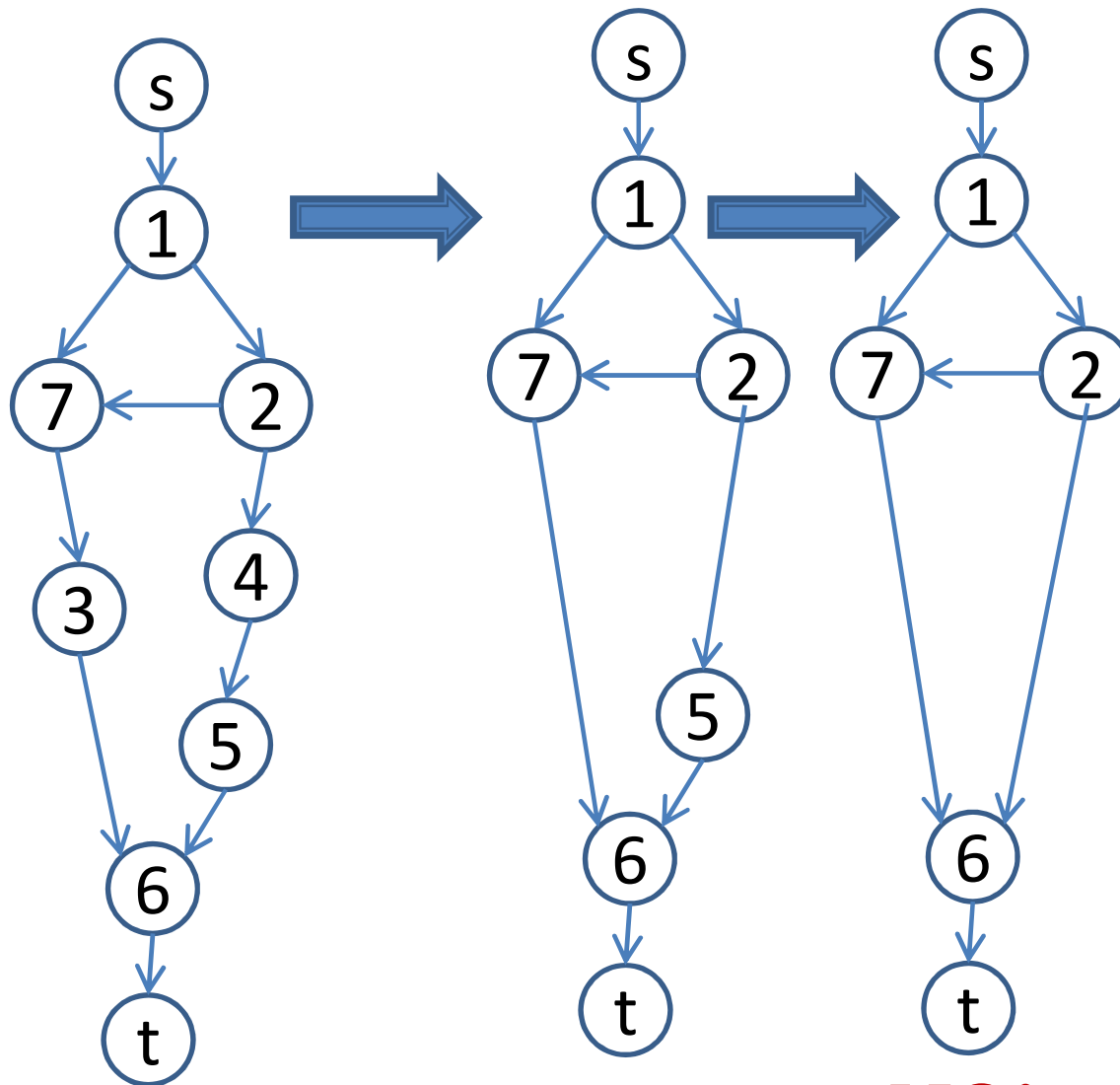
G is SP iff
 $\text{MaxRed}(G) = \text{BSP}$

- ▶ **MaxRed(G)**: iteratively performs series and parallel reductions on a given graph G

- ▶ **BSP**: Basic Series-Parallel



Is it Series-Parallel?

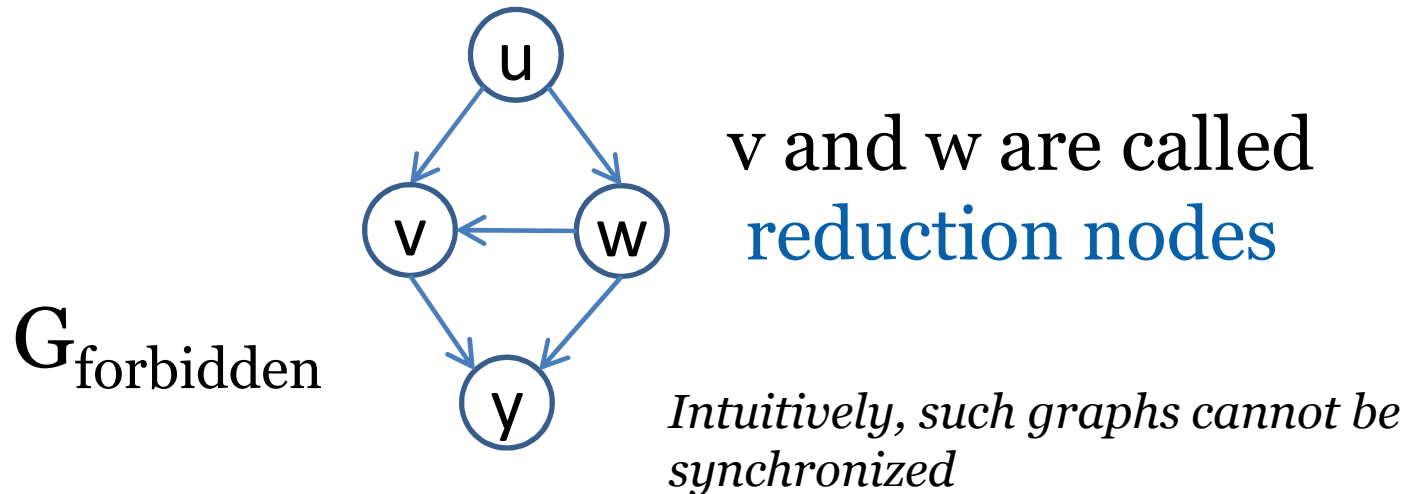


NO!

... Another definition of series-parallel graphs?

Another definition (Non SP-graphs)

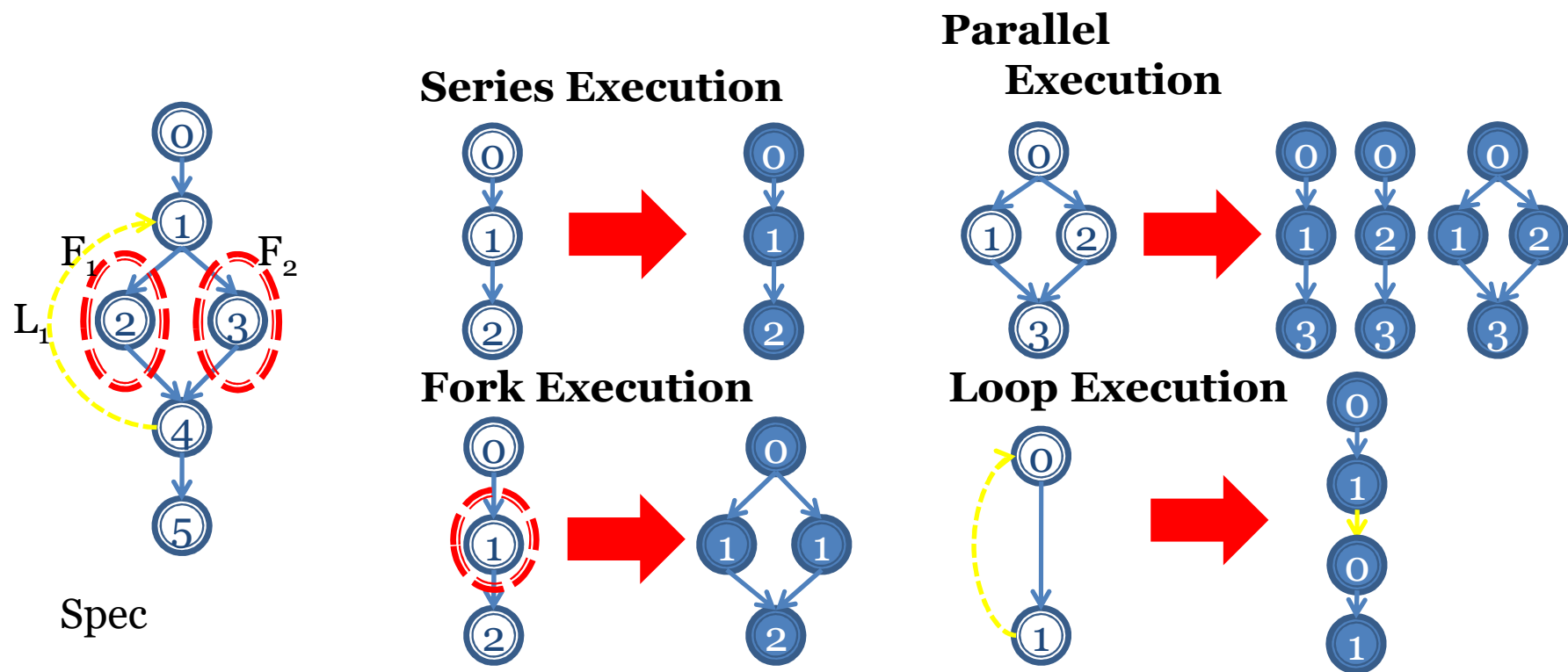
G is non-SP iff $\text{MaxRed}(G)$ contains $G_{\text{forbidden}}$



Subgraph isomorphism is polynomial for SP graphs

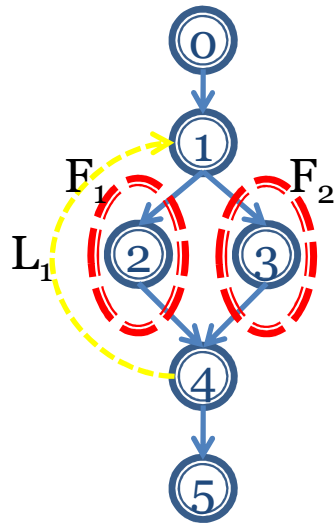
SPFL-Workflow Model (PDiffView)

- ▶ Workflow Specification
 - A series-parallel graph overlaid with well-nested fork and loop subgraphs
 - Four kinds of executions: series, parallel, fork and loop

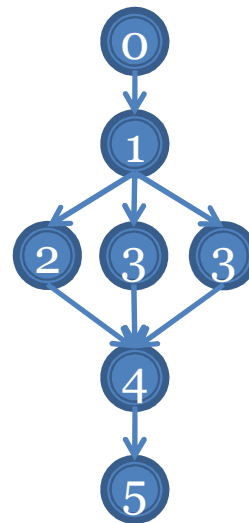


SPFL-Workflow Model

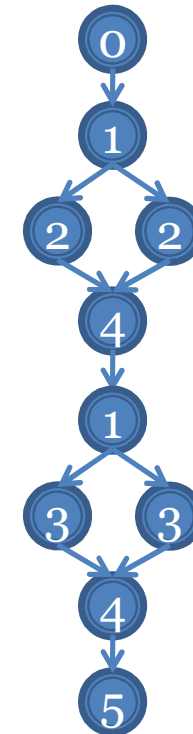
- ▶ Valid Runs
 - Derived from the specification by applying series, parallel, fork and loop executions recursively



Spec (G, F, L)



Valid run R_1

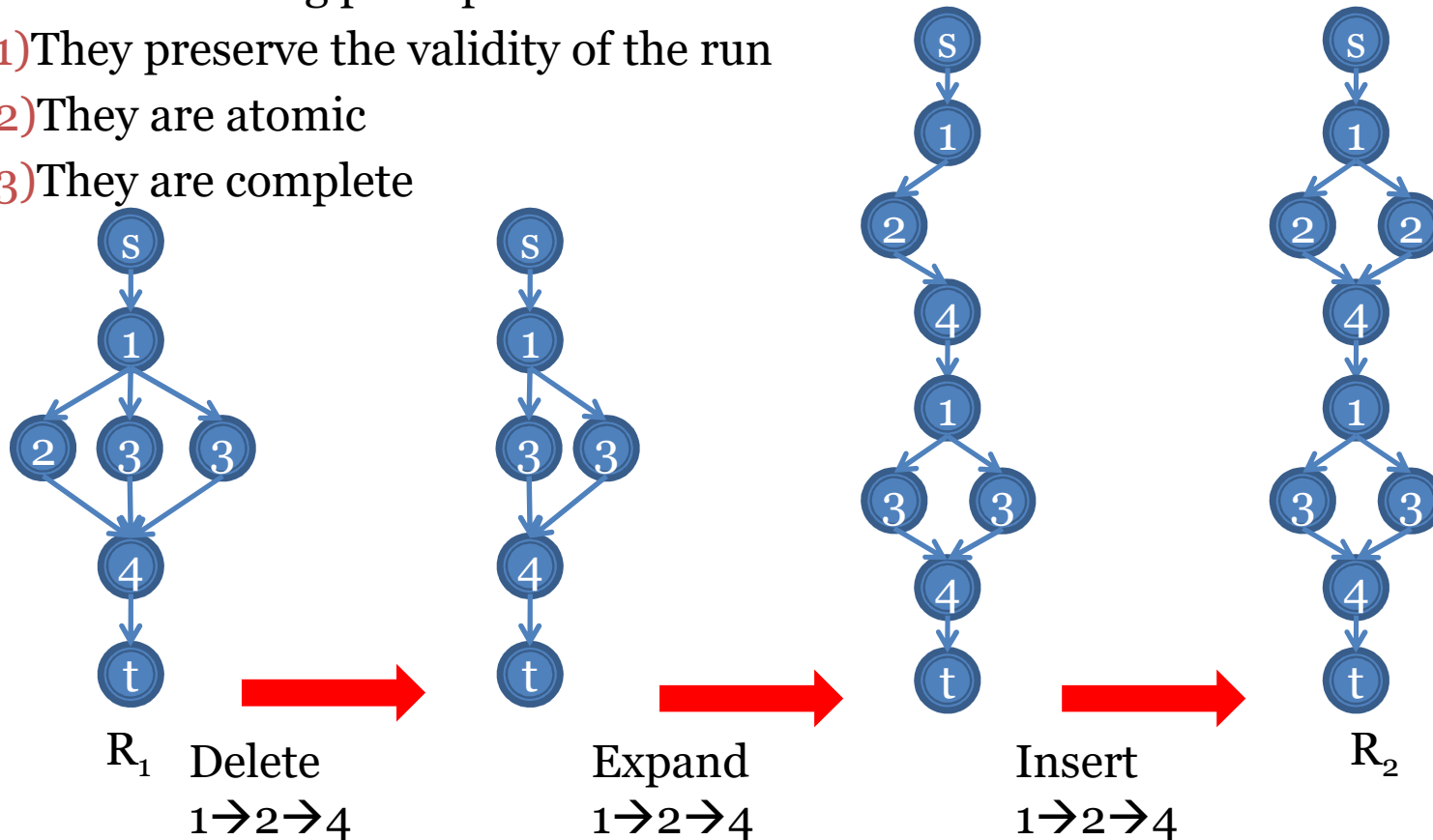


Valid run R_2

Edit Operations

▶ Path Insertion, Deletion, Expansion, Contraction

- **Elementary path**: each internal vertex has exactly one incoming edge and one outgoing edge, and the resulting graph is still valid with respect to the specification.
- Three motivating principles
 - (1) They preserve the validity of the run
 - (2) They are atomic
 - (3) They are complete

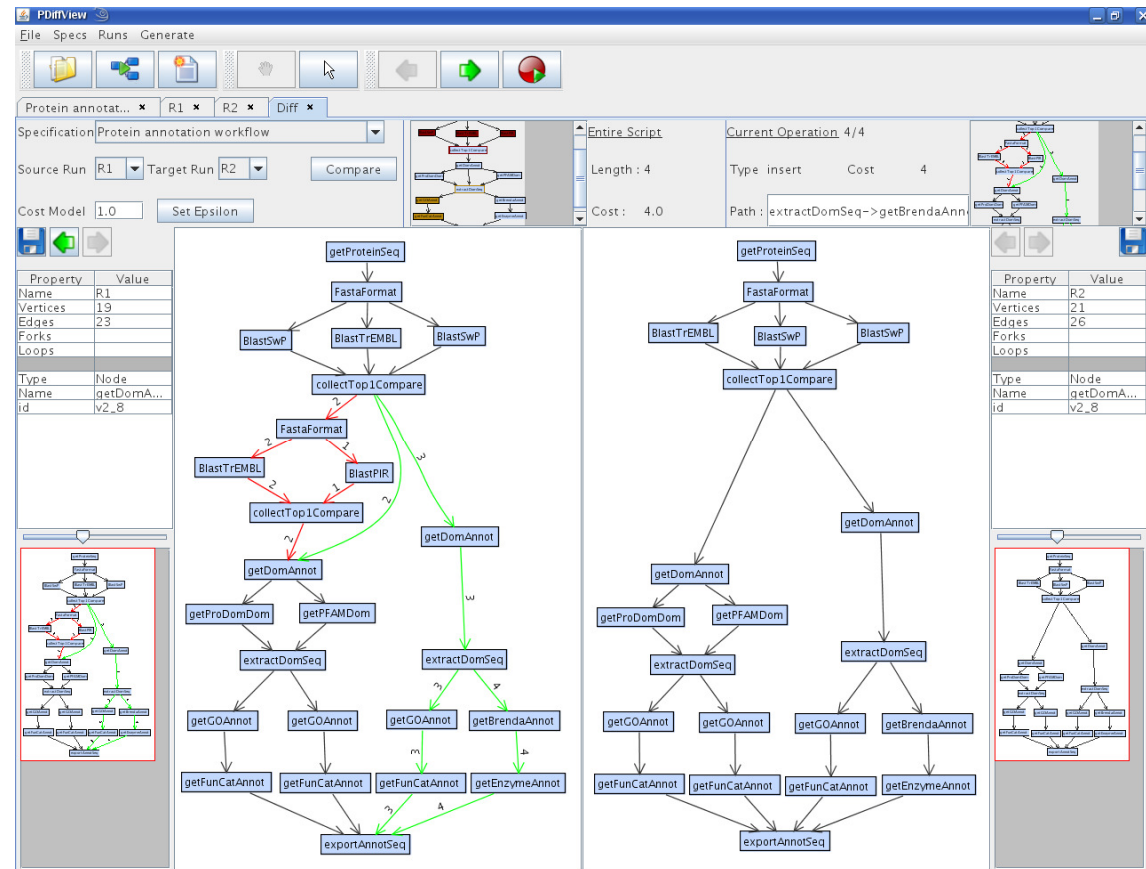


PDiffView

Polynomial-time algorithm designed in PDiffView for SPFL workflows

Problem statement

Given a pair of valid runs R_1 and R_2 of the same specification, and a cost function, compute a **minimum cost edit script** that transforms R_1 to R_2 . The cost of this edit script is also known as the **edit distance** between R_1 and R_2



Conclusion on workflows

- ▶ Workflows plays a crucial role in biological data integration
- ▶ Various areas of computer sciences are involved
 - Databases (e.g., to query and store them)
 - Software engineering (e.g., to optimize or rewrite them)
 - Graph algorithmics (e.g., to query and compare them)
 - ... and a lot of other optimization techniques
- ▶ Very large spectrum of challenges
 - From very theoretical (e.g., graph theory, equivalence of programs) to very technical and practical (user study, benchmarking on real data sets...)

This Tutorial

- ▶ Part I – Data Integration workflows
 - What are scientific workflow systems
 - Designing a workflow from scratch
 - Repositories of workflows and web services (reuse)
 - workflows and reproducibility
 - Latest results on workflows
 - Or How CS research may have direct impact on LS

- ▶ **Part II – Ranking Biological data**
 - Ranking criteria
 - Introducing ranking into integration solutions
 - Data warehouses
 - Portals

- ▶ Part III – Conclusions

Ranking bio data is needed!

- ▶ A few things provided in Entrez
 - By alphabetical order of ids, *relevance*... (Gene)
 - By date of publication... (Medline)
- ▶ Biological data have **specific features**
 - Data from sources **reflect expertise**
→ DBs are different (reliability etc.)
 - **Cross-references** are not just hypertext links
 - **Different qualities**: Manually provided or automatically obtained
 - **Different meanings**: More info can be found at, is very different from, is similar to...
 - Several **goals to achieve** when querying
 - The most famous data, the most reliable, the freshest...

Rankings for bio data in warehousing

▶ BioZon

- [Birkland *et al*, 2006], <http://www.biozon.org/>
- Graph-based approach (graph of entities as a support for queries)
- Variants of google Page-rank algorithm
- Difficulties
 - To be constantly updated...
 - Google-like (page-rank, object rank): [probability distribution](#) used to represent the likelihood that a person randomly clicking on links will arrive at any particular page.
 - Requires the knowledge of the entire graph « local pageRank »
 - All the sources of data have the same « value »

→ **Wanted**: Ranking solution exploiting links (several paths led to the same data) + reliability of the sources +...

- Problem : How to [combine](#) all such criteria?

▶ Alternative: Consensus rankings?

Consensus rankings

- ▶ Generating a *consensus ranking* to make the most of used ranking methods applied to biological data by
 - Putting emphasis on their *common points*
 - Not putting too much importance on data classified “good” by only one or *a few ranking* methods

Various consensus ranking techniques

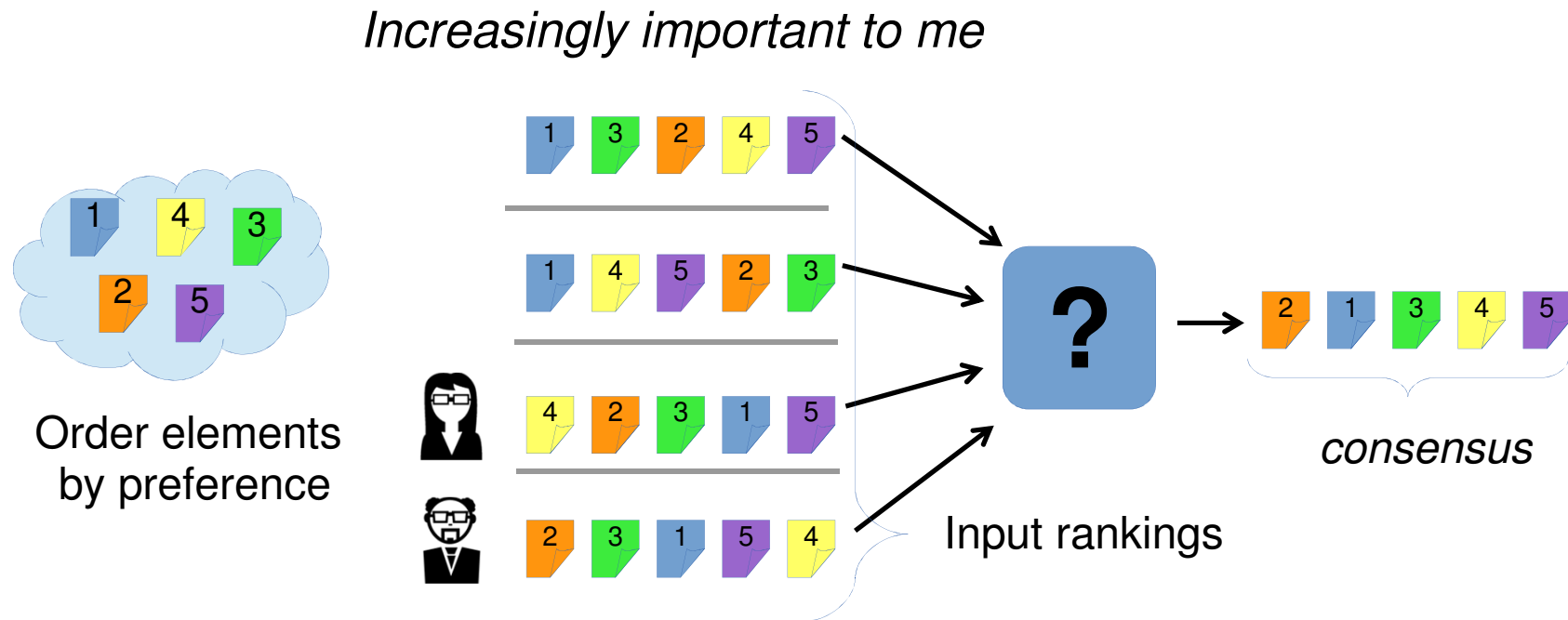
▶ Numerous applications

- Voting system [Borda 1781]
- Web information search [Dwork *et al.* 2001]
- Biological data search [DeConde *et al.* 2006]
- Agregation of opinions [Kittur *et al.* 2008]

▶ Numerous communities

- Sociales Sciences [Ali *et al.* 2012]
- Algorithmics [Ailon *et al.* 2008]
- Databases [Fagin *et al.* 2004]
- Biology [Sese *et al.* 2001]

Example



How to define a consensus ?
How to compute a consensus ?

Distances and algorithms

BioConsert [Cohen-Boulakia et al. 2011]

FaginDyn [Fagin et al. 2004]

Generalized
Kendall- τ distance

Ailon3/2 [Fagin et al. 2004]

RepeatChoice [Ailon et al. 2010]

Pick-A-Perm [Ailon et al. 2008]

KwikSort [Ailon et al. 2008]

PNE (exact) [Conitzer, et al. 2006]
(sans égalité entre éléments)

B&B [Ali et al. 2012]

Chanas [Chanas et al. 1996]

MEDRank [Fagin et al. 2003]

BordaCount [Borda 1781]

CopelandMethod [Copeland et al. 1951]

MC4 [Dwork et al. 2001]

Positionnal approaches

ChanasBoth [Coleman et al. 2009]

Kendall- τ distance

97/45

Kendall- τ [Kendall 1938] Distance

« Find a consensus close to input rankings »

The Kendall- τ $D(\pi, \sigma)$ distance counts the number of pairs of elements inversed (ie in the opposite order) between two rankings.

$$D(\pi, \sigma) = \left| \left\{ (i, j) : i < j \wedge \left(\begin{array}{l} \pi[i] < \pi[j] \wedge \sigma[i] > \sigma[j] \\ \vee \pi[i] > \pi[j] \wedge \sigma[i] < \sigma[j] \end{array} \right) \right\} \right|$$

$$\begin{aligned} \pi_1 &:= [A, D, C, B] \\ \pi_2 &:= [B, A, D, C] \end{aligned}$$

$$D(\pi_1, \pi_2) =$$



Kendall- τ [Kendall 1938] Distance

« Find a consensus close to input rankings »

The Kendall- τ $D(\pi, \sigma)$ distance counts the number of pairs of elements inversed (ie in the opposite order) between two rankings.

$$D(\pi, \sigma) = \left| \left\{ (i, j) : i < j \wedge \left(\begin{array}{l} \pi[i] < \pi[j] \wedge \sigma[i] > \sigma[j] \\ \vee \pi[i] > \pi[j] \wedge \sigma[i] < \sigma[j] \end{array} \right) \right\} \right|$$

$$\begin{aligned} \pi_1 &:= [A, D, C, B] \\ \pi_2 &:= [B, A, D, C] \end{aligned}$$

$$\begin{aligned} D(\pi_1, \pi_2) &= 1_{A>B} \\ &\quad + 1_{B>D} \\ &\quad + 1_{B>C} \\ &= 3 \end{aligned}$$

Optimal consensus

Kemeny Score

$$S(\pi, \mathcal{P}) = \sum_{\sigma \in \mathcal{P}} D(\pi, \sigma)$$

Optimal Consensus

$$\forall \pi \in \mathcal{S}_n : S(\pi^*, \mathcal{P}) \leq S(\pi, \mathcal{P})$$

Complexity [Dwork *et al* 2001, Biedl *et al.* 2009]

NP-Difficult for an odd number of permutations ≥ 4

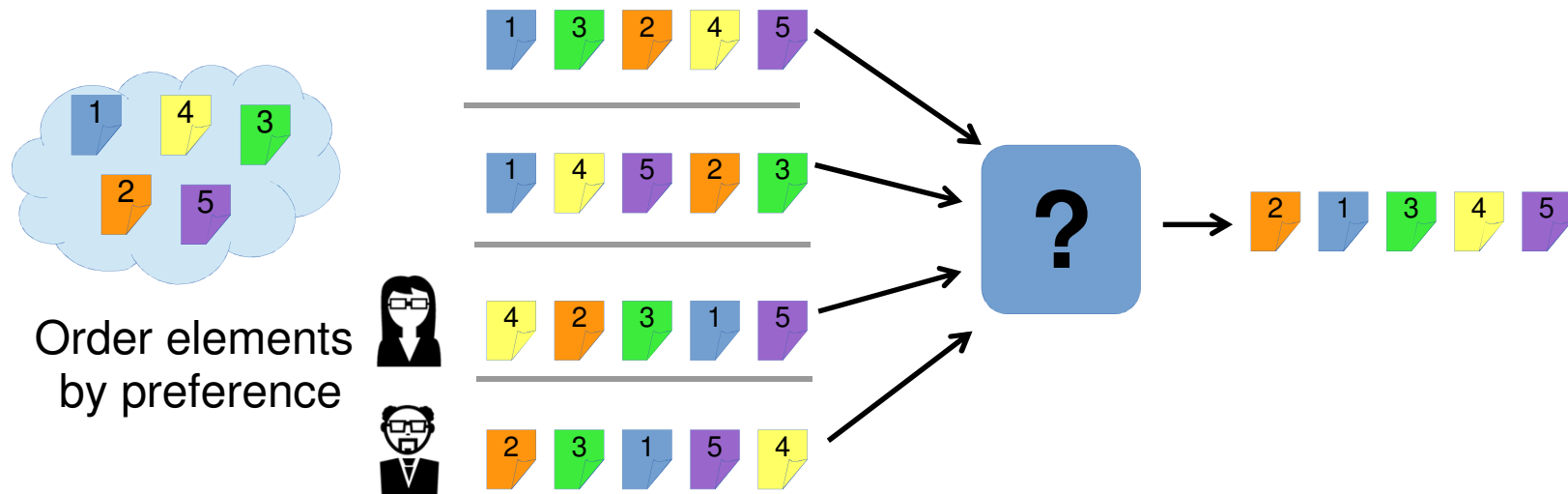
$$\mathcal{P} \left\{ \begin{array}{l} \pi_1 = [A, D, C, B] \\ \pi_2 = [B, A, D, C] \\ \pi_3 = [D, A, B, C] \end{array} \right.$$

$$\pi^* = [A, D, B, C]$$

$$\begin{aligned} S(\pi^*, \mathcal{P}) &= 1_{A>B@ \pi_2} \\ &\quad + 1_{A>D@ \pi_3} \\ &\quad + 1_{B>C@ \pi_1} \\ &\quad + 1_{B>D@ \pi_2} \\ &= 4 \end{aligned}$$

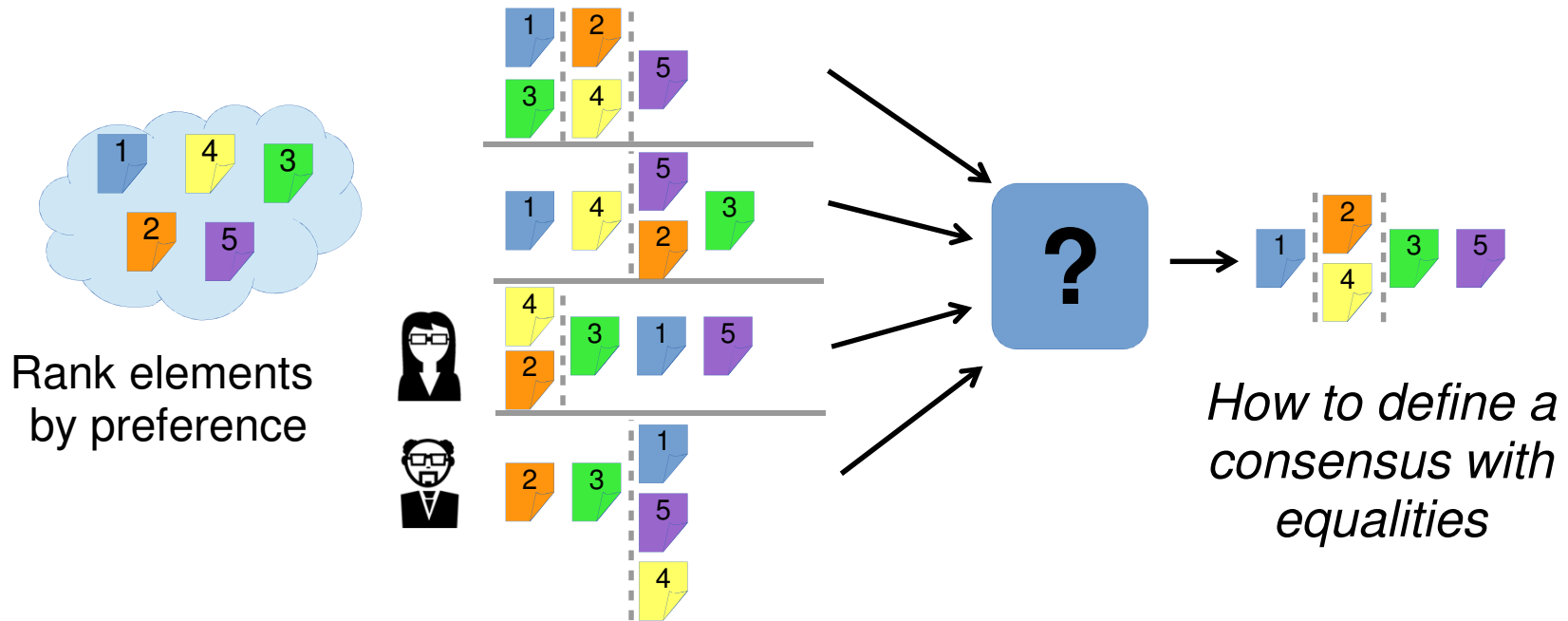
Back to real data...

Increasingly important to me



Real data with equalities

Increasingly important to me



Generalized Kendall- τ [Fagin et al. 2004]

« Find a consensus close to input rankings »

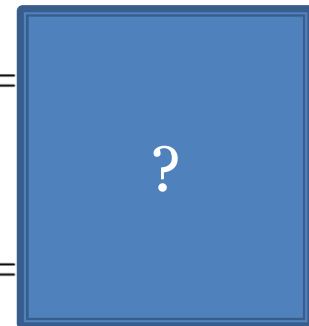
Generalized Kendall- τ $G(r,s)$ counts the number of pairs of elements

- inversed between two rankings r et s
- tied in only one of the two rankings

$$G(r, s) = \left| \left\{ \begin{array}{l} (i, j) : i < j \wedge \\ \left(\begin{array}{l} r[i] < r[j] \wedge s[i] > s[j] \\ \vee r[i] > r[j] \wedge s[i] < s[j] \\ \vee r[i] = r[j] \wedge s[i] \neq s[j] \\ \vee r[i] \neq r[j] \wedge s[i] = s[j] \end{array} \right) \end{array} \right\} \right|$$

$$r_1 = [\{A\}, \{D\}, \{B, C\}]$$
$$r_2 = [\{B\}, \{A\}, \{D\}, \{C\}]$$

$$G(r_1, r_2) =$$



Generalized Kendall- τ [Fagin et al. 2004]

« Find a consensus close to input rankings »

Generalized Kendall- τ $G(r,s)$ counts the number of pairs of elements

- inversed between two rankings r et s
- tied in only one of the two rankings

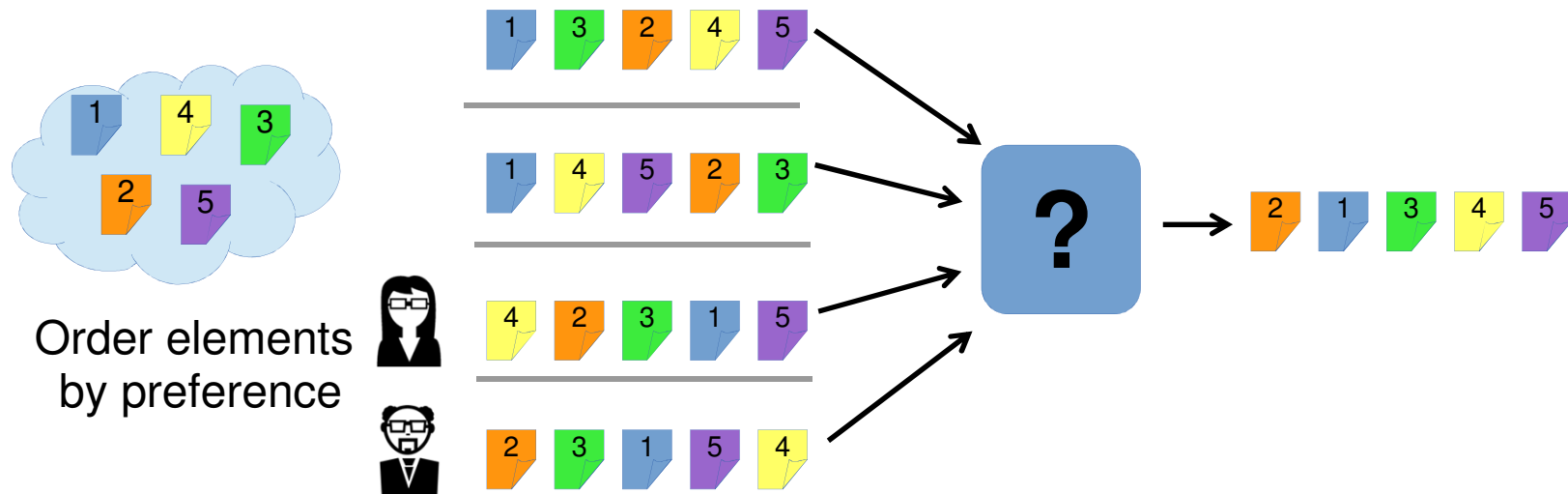
$$G(r, s) = \left| \left\{ \begin{array}{l} (i, j) : i < j \wedge \\ \left(\begin{array}{l} r[i] < r[j] \wedge s[i] > s[j] \\ \vee r[i] > r[j] \wedge s[i] < s[j] \\ \vee r[i] = r[j] \wedge s[i] \neq s[j] \\ \vee r[i] \neq r[j] \wedge s[i] = s[j] \end{array} \right) \end{array} \right\} \right|$$

$$\begin{aligned} r_1 &= [\{A\}, \{D\}, \{B, C\}] \\ r_2 &= [\{B\}, \{A\}, \{D\}, \{C\}] \end{aligned}$$

$$\begin{aligned} G(r_1, r_2) &= 1_{A>B@r_2} \\ &\quad + 1_{B>D@r_2} \\ &\quad + 1_{B=C@r_2} \\ &= 3 \end{aligned}$$

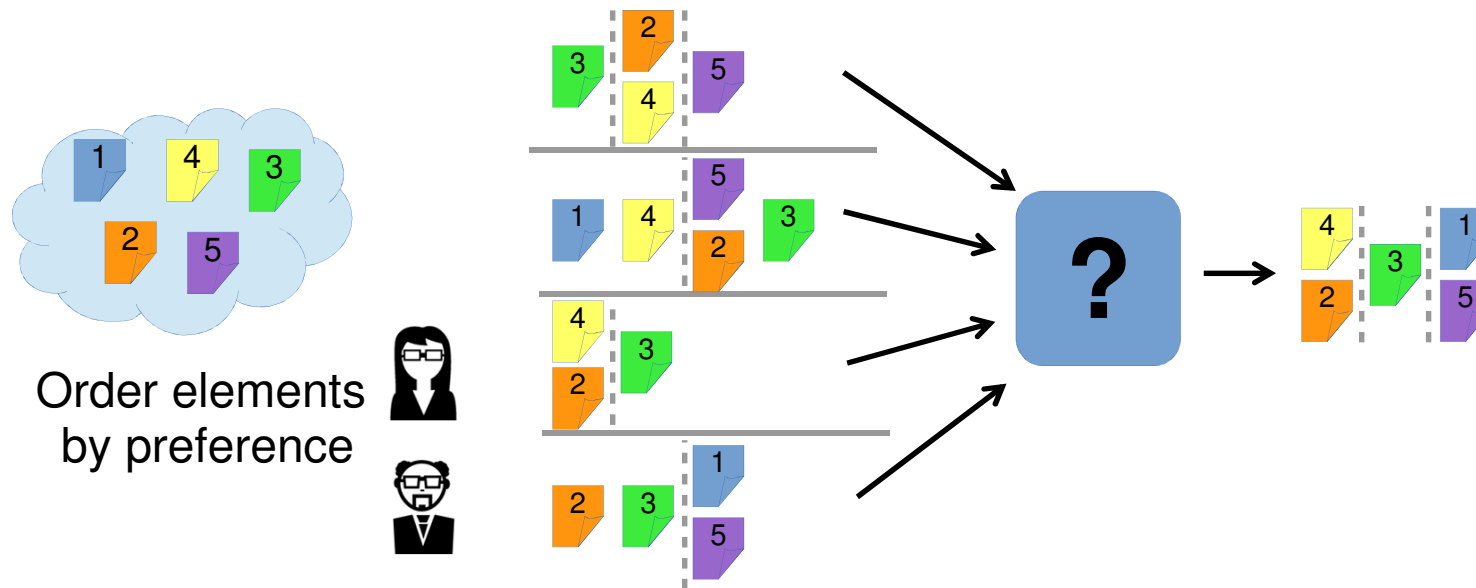
Back to real data...

Increasingly important to me

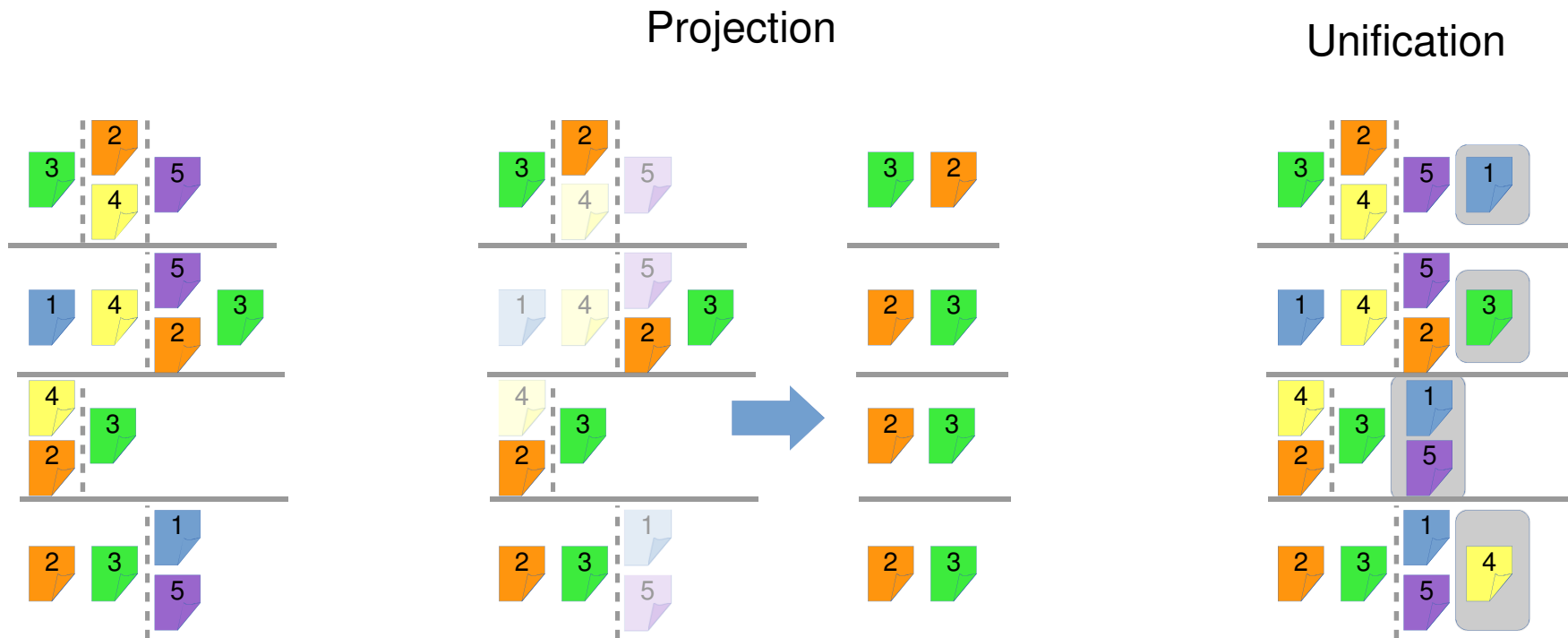


Real data: Incompleteness

Increasingly important to me



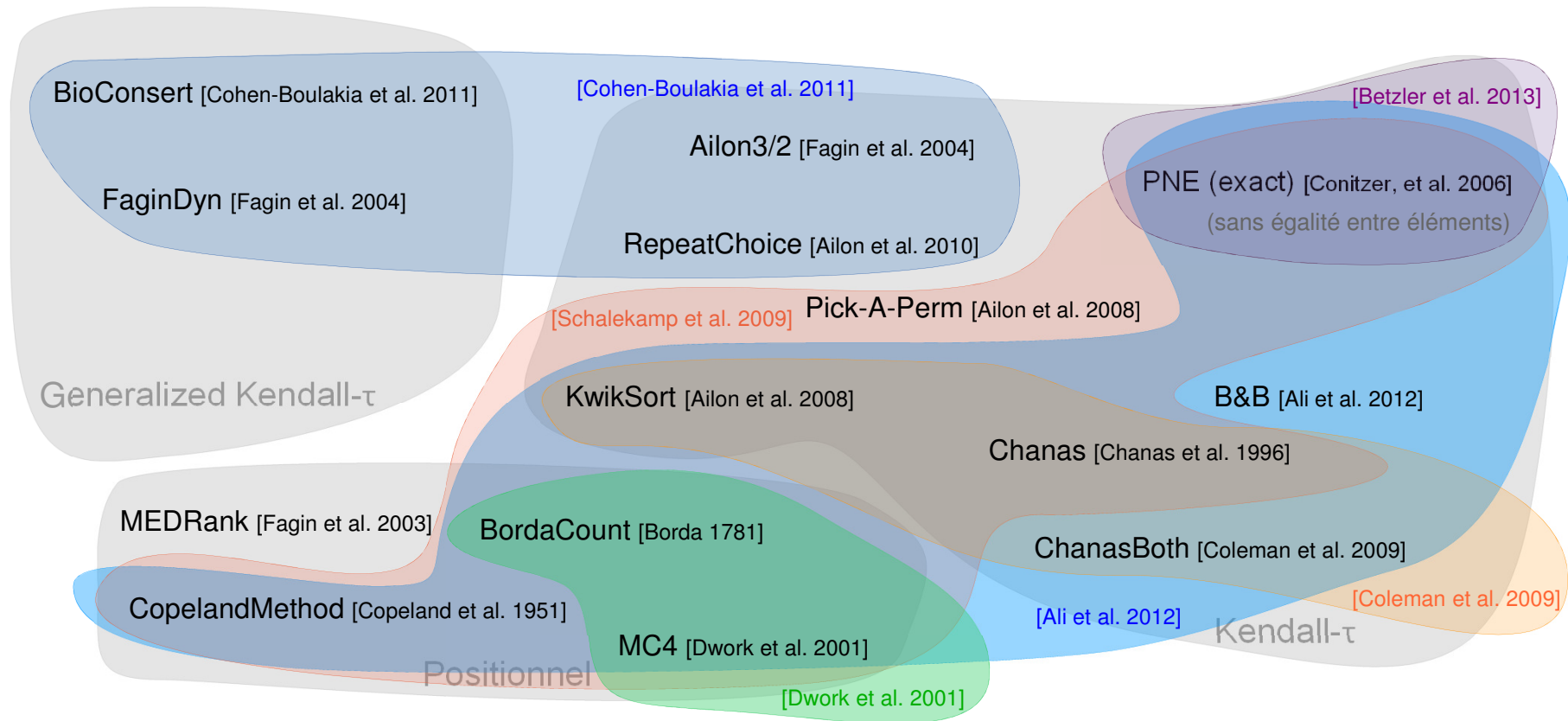
Normalizing a data set



Unrelevant data elements
are removed

Unrelevant data elements
are placed at the end of
one dedicated bucket

Previous studies: algorithms



Previous studies: data sets

Nom et publication		Sur les même éléments	n	m	#
EachMovie	CW09	non	100		146
F1	BBN13	projeté	[9; 28]	[11; 19]	39
BioMedical	CBDH11	unifié	[15; 402]	4	11
GiantSlalom	AM12	unifié fragmenté	59	16	1
SkiCross/Jumping	BBN13	projeté	69	4	1
WebCommunities	CW09; SZ09	oui	100	9	50
WebSearch	Dwo+01	–		7	37
	SZ09	unifié	$\bar{n} = 283$	4	37
	AM12	unifié	[275; 348]	4	37
	BBN13	projeté	[18; 163]	4	37
Mallows model	AM12	oui	{10, 50}	{100, 5000}	8
Mallows model	BBN13	oui	[10; 200]	[4; 20]	8400
Placket-Luce model	AM12	oui	{10, 50}	100	2
	BBN13		[10; 200]	[4; 20]	2240
RandomGraph	DK04	oui	[15 – 50]	[5; 35]	
RandomGraph	CW09	oui			1500
Random	AM12	oui	100	100	1
Random	CBDH11	oui	[4; 8]	4	2500

Similarity levels

- A few data sets reused
- Most of the data sets are not publicly available
- Different normalization methods used

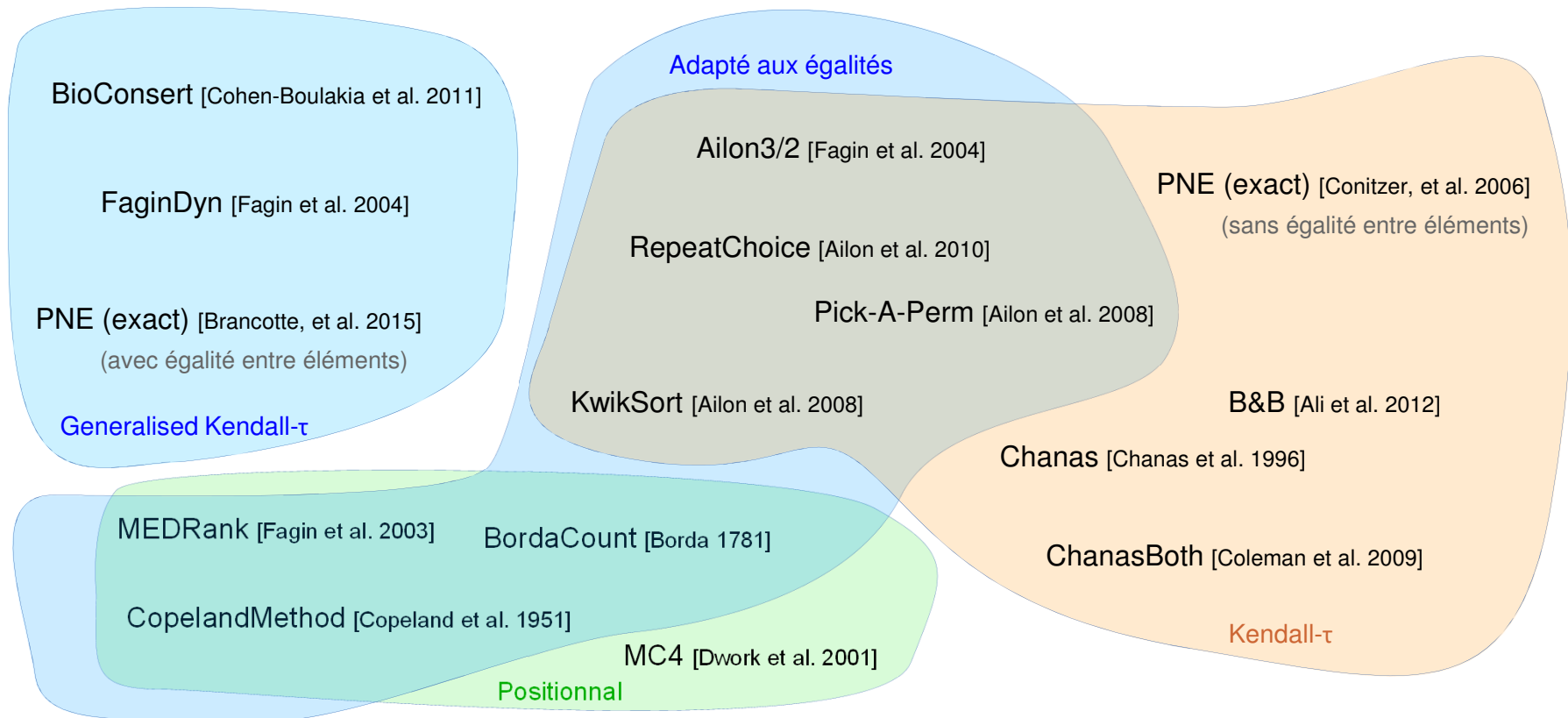
Results

- Various algorithms considered in each study, different normalisations, différent data sets
- Incomplete results, sometimes even contradictory
- Equalities are not considered
- Same behaviour of algorithms with equalities??
- Impact of similarity between data sets?



Need to compare approaches in a more systematic and exhaustive way!

Classifications of consensus algorithms



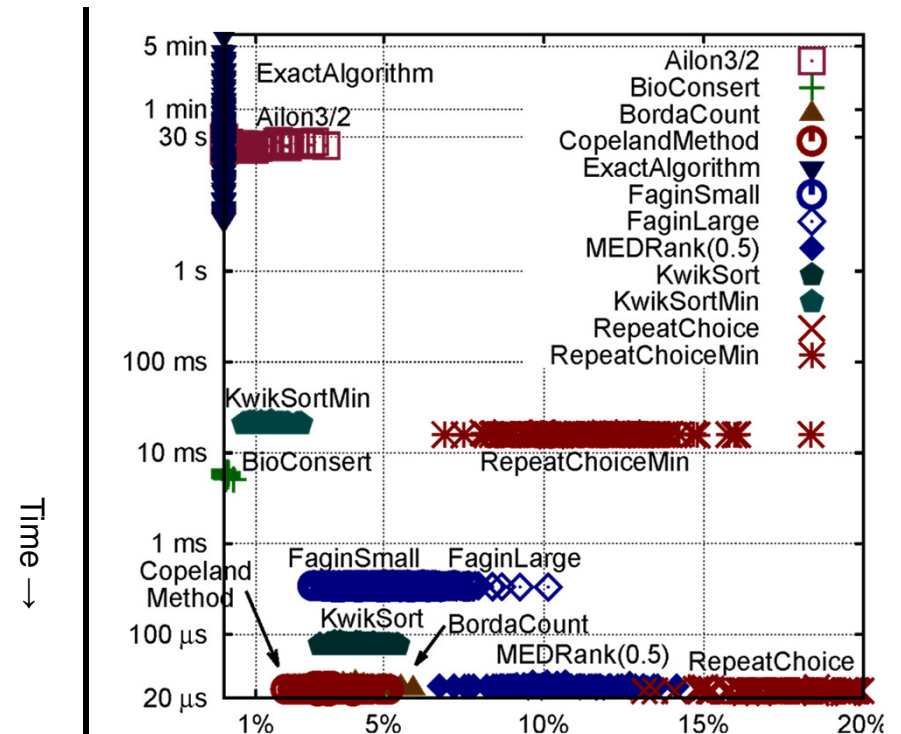
Classifications of consensus algorithms

Name	Approx.	Algorithm class	Can produce ties	Untying cost
Ailon ^{$\frac{3}{2}$}	3/2	[K] Linear Prog.	with slight modification	with slight modif.
BioConsert	2	[G] Local search	yes	yes
BordaCount	5	[P] Sort by score	with slight modification	no
Chanas	no	[K] Local search	no	—
ChanasBoth	no	[K] Local search	no	—
BnB	exact	[K] Branch & Bound	no	—
CopelandMethod	no	[P] Sort by score	with slight modification	no
FaginDyn	4	[G] Dynamic Prog.	yes	yes
ILP	exact	[K] Linear Prog.	only in input	with large modif.
KwikSort	$\frac{11}{7}$	[K] Divide & conquer	with slight modification	with slight modif.
MC4	no	[P] Hybrid	yes	no
MEDRank	no	[P] Extract order	with slight modification	no
Pick-a-Perm	2	[K] Naive	yes	—
RepeatChoice	2	[K] Sort by order	with slight modification	no

Recommandations

- **BioConsert** can be used in a very large majority of the cases
- For very large data sets (>30.000 elements)
 - **KwikSort** can be preferred
- If there is a need to seed up then
 - In case of few equalities use **BordaCount**
 - Otherwise use **MEDRank**
 - Alternativeley: use both algorithms and pick the best

<http://rank-aggregation-with-ties.lri.fr/>



High quality ← → low quality
(gap)

ConQuR-Bio: consensus of reformulations

Query NCBI so that equivalent queries provide the same results

Equivalent reformulations:
cervix cancer vs cervical cancer
(460 vs 20 genes)

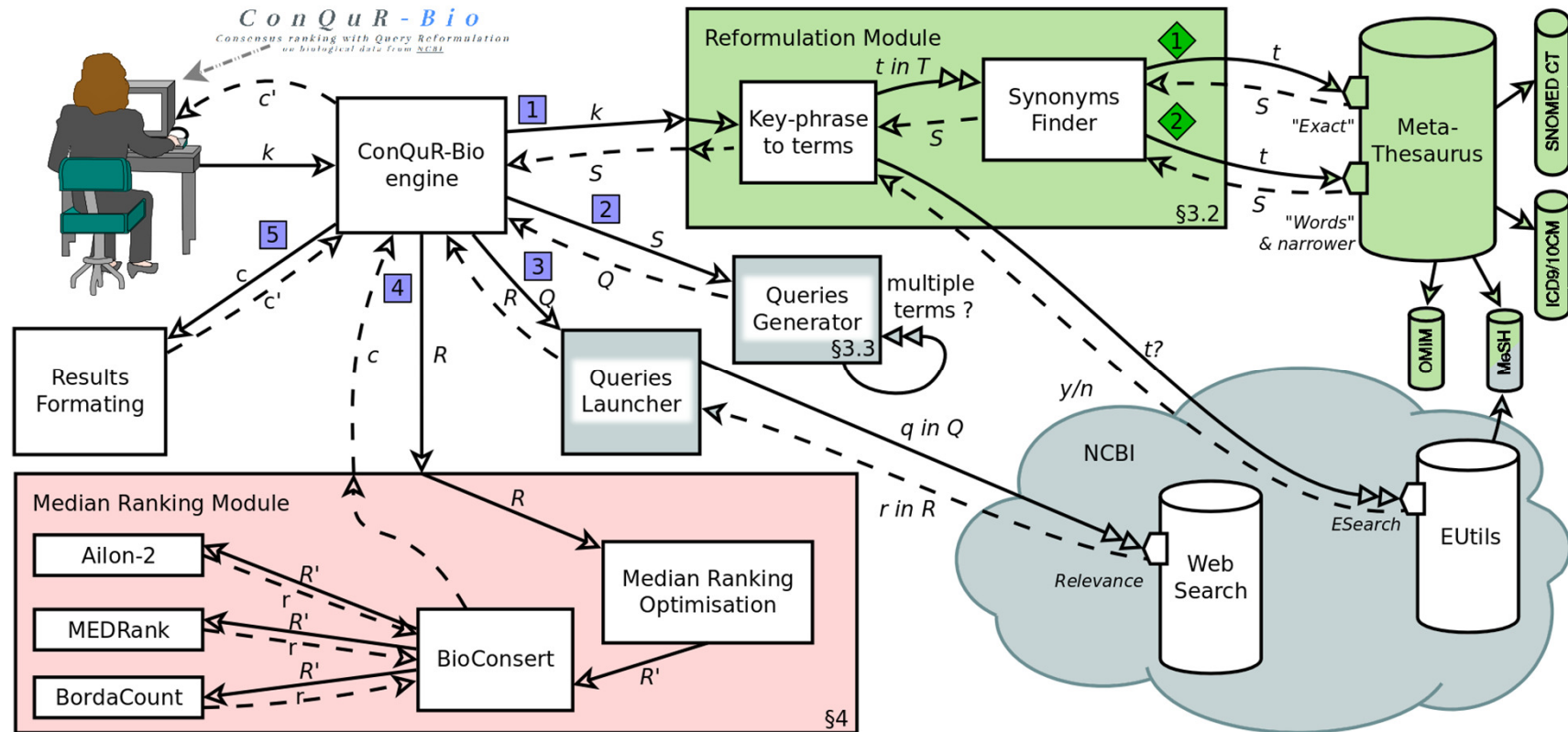
Abbreviations:
Attention deficit hyperactivity disorders vs ADHD
(109 vs 144 genes, 74 in common)

Linguistics variations:
tumour vs tumor
(& breast cancer) : 681 vs 291 genes

More precise reformulations :
colorectal cancer vs Lynch syndrom
(+6 new genes)

- ▶ Finding all reformulations is time-consuming
- ▶ Querying using all reformulations provide huge amounts of data sets which have to be ranked....

Architecture of ConQuR-Bio



I) Reformulations using 5 biomedical terminologies

II) Querying NCBI to get genes ranked by „relevance“

III) Aggregating using a series of consensus algorithms with a variant of the Generalized Kendall- τ distance

Conclusion on ranking

- ▶ Faced with the number of results obtained as answer to a query **ranking results is crucial**
 - **Prioritize experiments**
- ▶ Very important data may be in relatively small DB
- ▶ Bad quality data may be highly referenced...
- ▶ Various ranking criteria can be taken into account
 - Freshness, Reliability, Completeness...
- ▶ Combining criteria is difficult
 - Consensus rankings provide good solutions
 - Expensive (time) → optimisation techniques needed

Still a lot to do!

Conclusions

- ▶ Data Integration in the Life Sciences is a hot topic where progress in research and Engineering have a mutual impact
- ▶ Reproducibility and data quality are key points
- ▶ The project of this module should help you understand the problems by yourself... and challenge the current solutions!