Facilitating complex life science data integration and reuse

Olivier Dameron

Université de Rennes 1

06 November 2019

Life science data

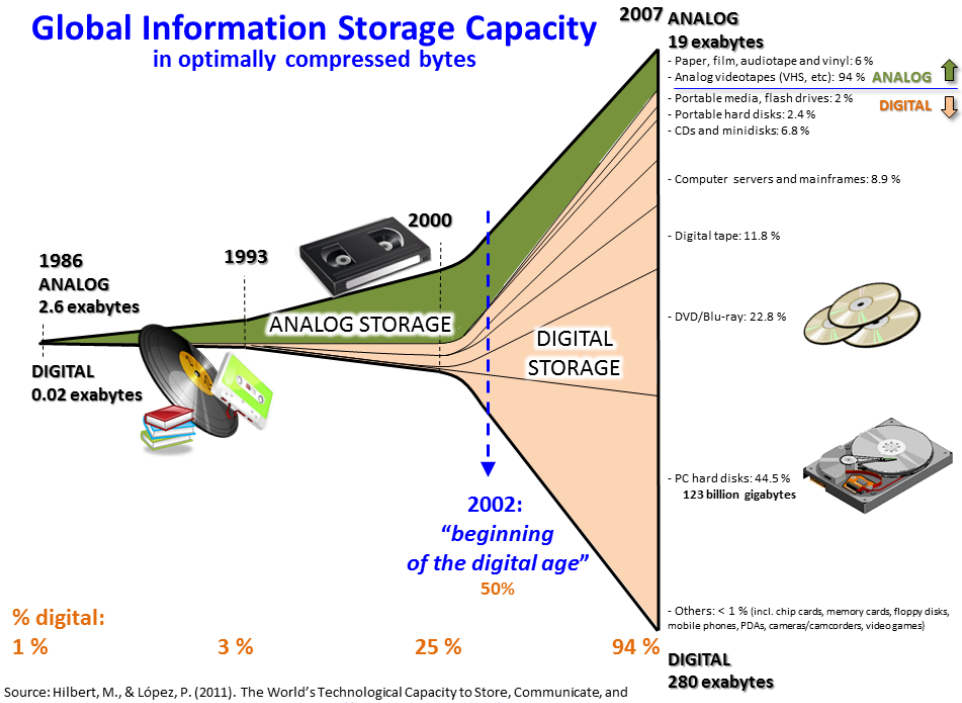
(Big) data science

Data science [Naur1974, Cleveland2001]

Extracting knowledge from (un)structured data

- **Numeric data** $\rightarrow$ statistics and deep learning
- **Symbolic data** $\rightarrow$ deductive reasoning, IA

Computerized data $\Rightarrow$ Systematic and automatic data processing



Source: Hilbert, M., & López, P. (2011). The World's Technological Capacity to Store, Communicate, and

Big data and the deluge of life science data

Big data

Datasets so **large** or **complex** that traditional data processing is inadequate [Laney2001]

Life science : data deluge [Aldhous1993]

- computerized biomedical data
- genomics and bioinformatics

Science. 1993 Oct 22;262(5133):502-3.

Managing the genome data deluge.

Aldhous P.

PMID: 8211171 [PubMed - indexed for MEDLINE]

Science. 1995 Aug 4;269(5224):630.

Europe opens institute to deal with gene data deluge.

Williams N.

PMID: 7624788 [PubMed - indexed for MEDLINE]

Bottleneck

Too much data for current processing capabilities

- data production rates outpace CPU improvements
- current analysis methods do not scale up

The Widening Gulf between Genomics Data Generation and Consumption: A Practical Guide to Big Data Transfer Technology

Frank A. Feltus[1], Joseph R. Breen III[2], Juan Deng[3], Ryan S. Izard[3], Christopher A. Konger[4], Walter B. Ligon III[3], Don Preuss[5] and Kuang-Ching Wang[3]

BIOINFORMATICS AND BIOLOGY INSIGHTS 2015:9(S1)

What to expect for 2025 ?

Our estimation is that genomics is a "four-headed beast" – it is either on par with or the most demanding domain [...] in terms of

- data acquisition
- data storage
- data distribution
- data analysis

Big Data: Astronomical or Genomical?

Zachary D. Stephens[1], Skylar Y. Lee[1], Faraz Faghri[2], Roy H. Campbell[2], Chengxiang Zhai[3], Miles J. Efron[4], Ravishankar Iyer[1], Michael C. Schatz[5]*, Saurabh Sinha[3]*, Gene E. Robinson[6]*

PLOS Biology | DOI:10.1371/journal.pbio.1002195 July 7, 2015

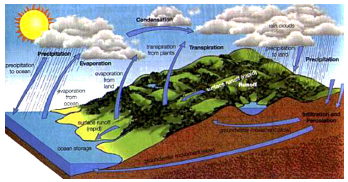
Table 1. Four domains of Big Data in 2025. In each of the four domains, the projected annual storage and computing needs are presented across the data lifecycle.

Data Phase	Astronomy	Twitter	YouTube	Genomics
Acquisition	25 zetta-bytes/year	0.5–15 billion tweets/year	500–900 million hours/year	1 zetta-bases/year
Storage	1 EB/year	1–17 PB/year	1–2 EB/year	2–40 EB/year
Analysis	In situ data reduction	Topic and sentiment mining	Limited requirements	Heterogeneous data and analysis
	Real-time processing	Metadata analysis		Variant calling, ~2 trillion central processing unit (CPU) hours
	Massive volumes			All-pairs genome alignments, ~10,000 trillion CPU hours
Distribution	Dedicated lines from antennae to server (600 TB/s)	Small units of distribution	Major component of modern user's bandwidth (10 MB/s)	Many small (10 MB/s) and fewer massive (10 TB/s) data movement

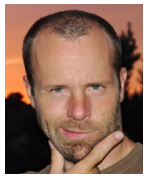
doi:10.1371/journal.pbio.1002195.t001

Complexity of life science data : (1) multiple scales

Ecosystem

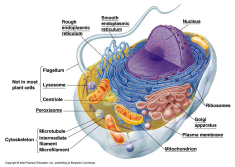


Organism

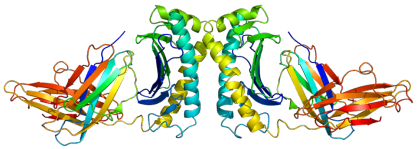


Organ

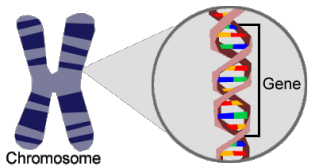
Cell



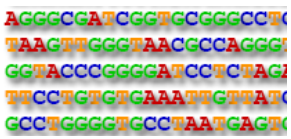
Protein



Gene



Sequence



Complexity of life science data : (2) (explicit) interdependence at each level

Ecosystem

Organism

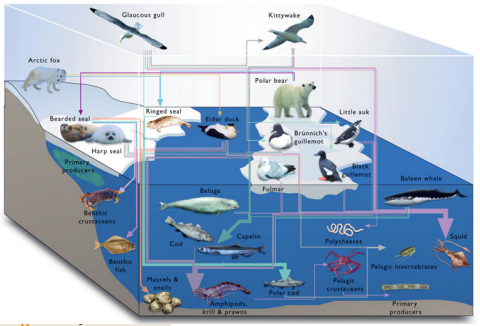
Organ

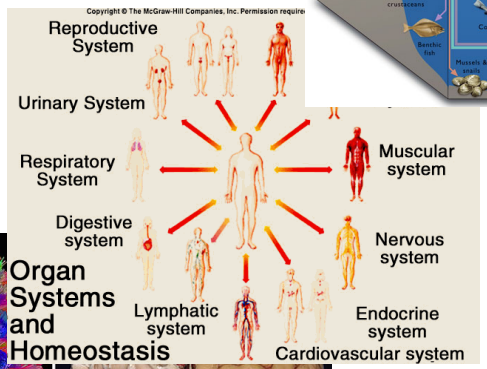
Cell

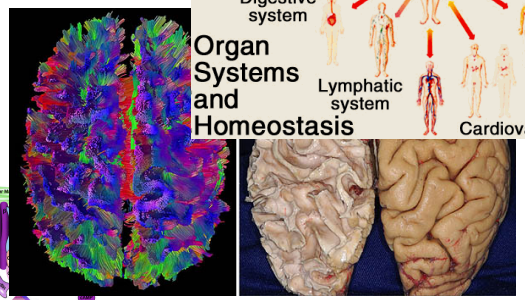
Protein

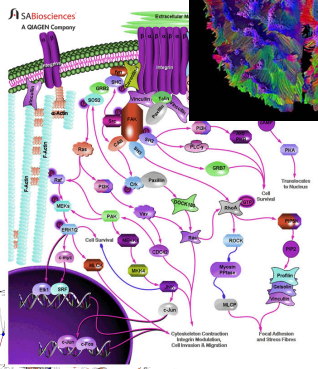
Gene

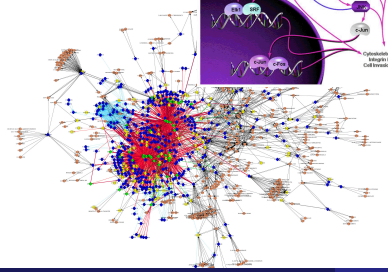
Sequence











Complexity of life science data : (3) scale (implicit) interdependence

Ecosystem

Organism

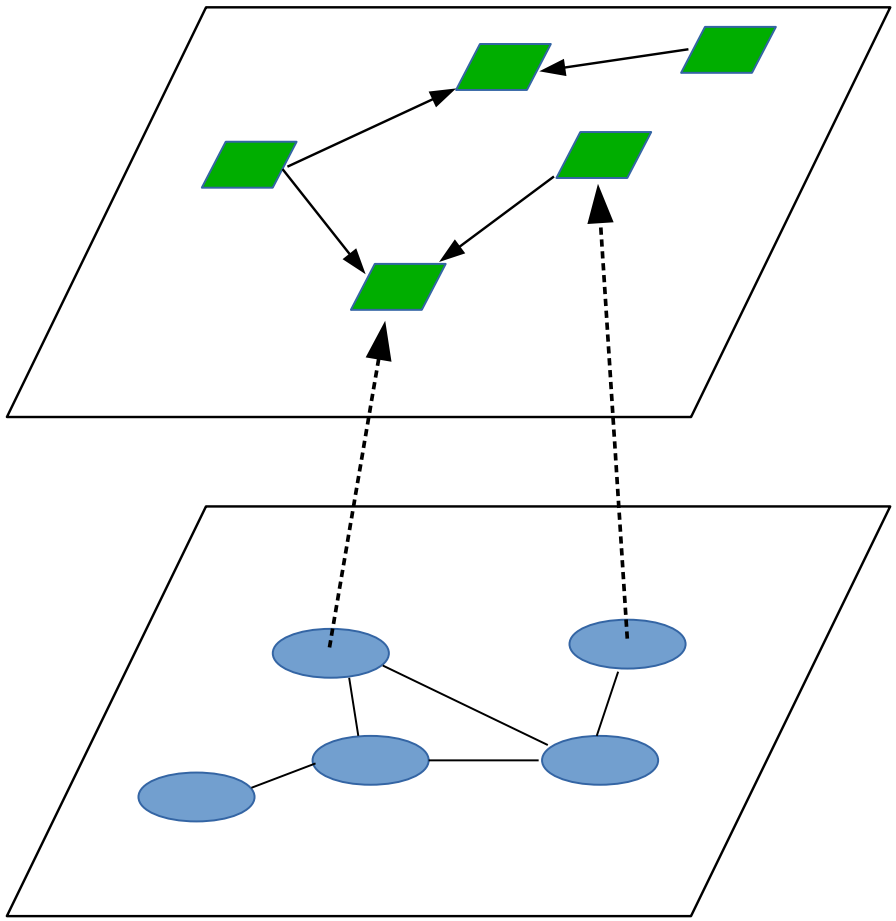
Organ

Cell

Protein

Gene

Sequence



Complexity of life science data : (3) scale (implicit) interdependence

Ecosystem

Organism

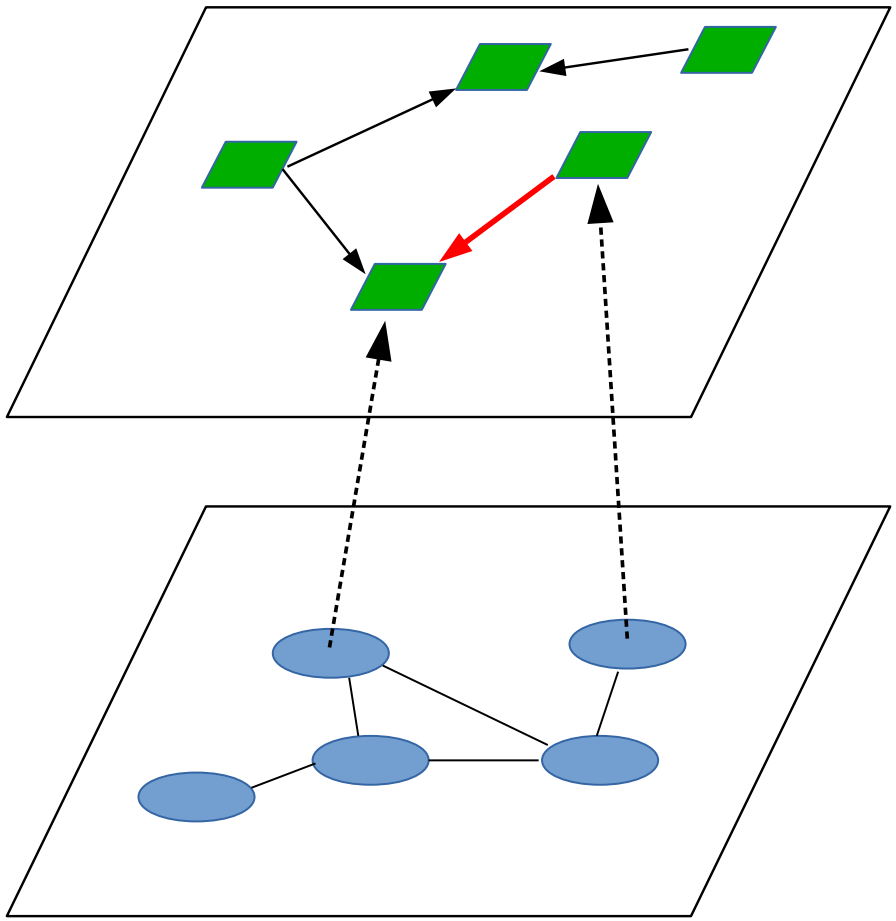
Organ

Cell

Protein

Gene

Sequence



Complexity of life science data : (3) scale (implicit) interdependence

Ecosystem

Organism

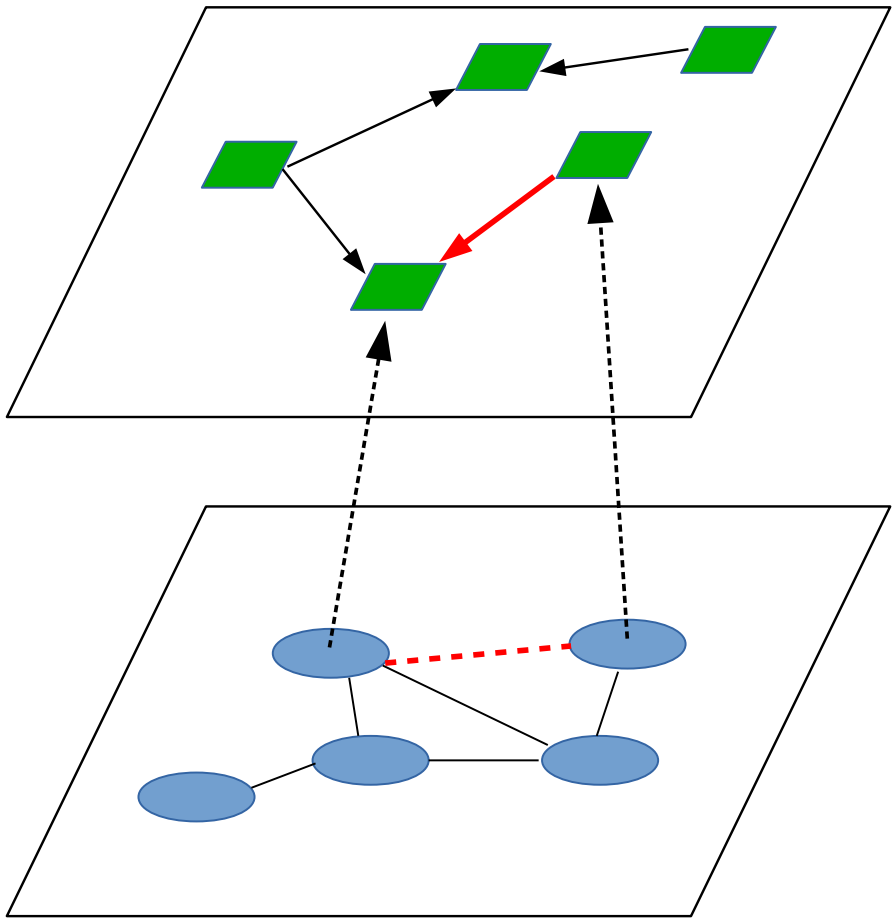
Organ

Cell

Protein

Gene

Sequence



Complexity of life science data : (4) variability

Ecosystem

Organism

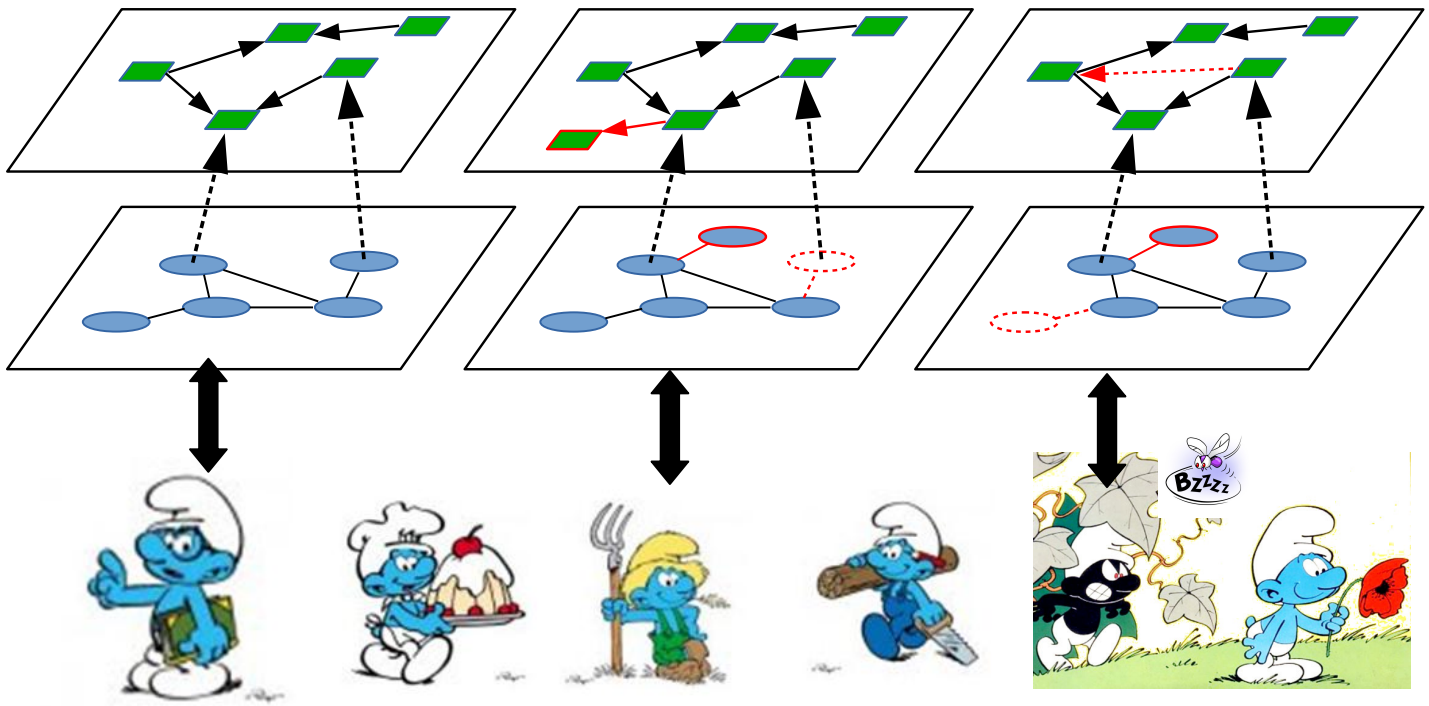
Organ

Cell

Protein

Gene

Sequence



Complexity of life science data : (5) incompleteness

Ecosystem

Organism

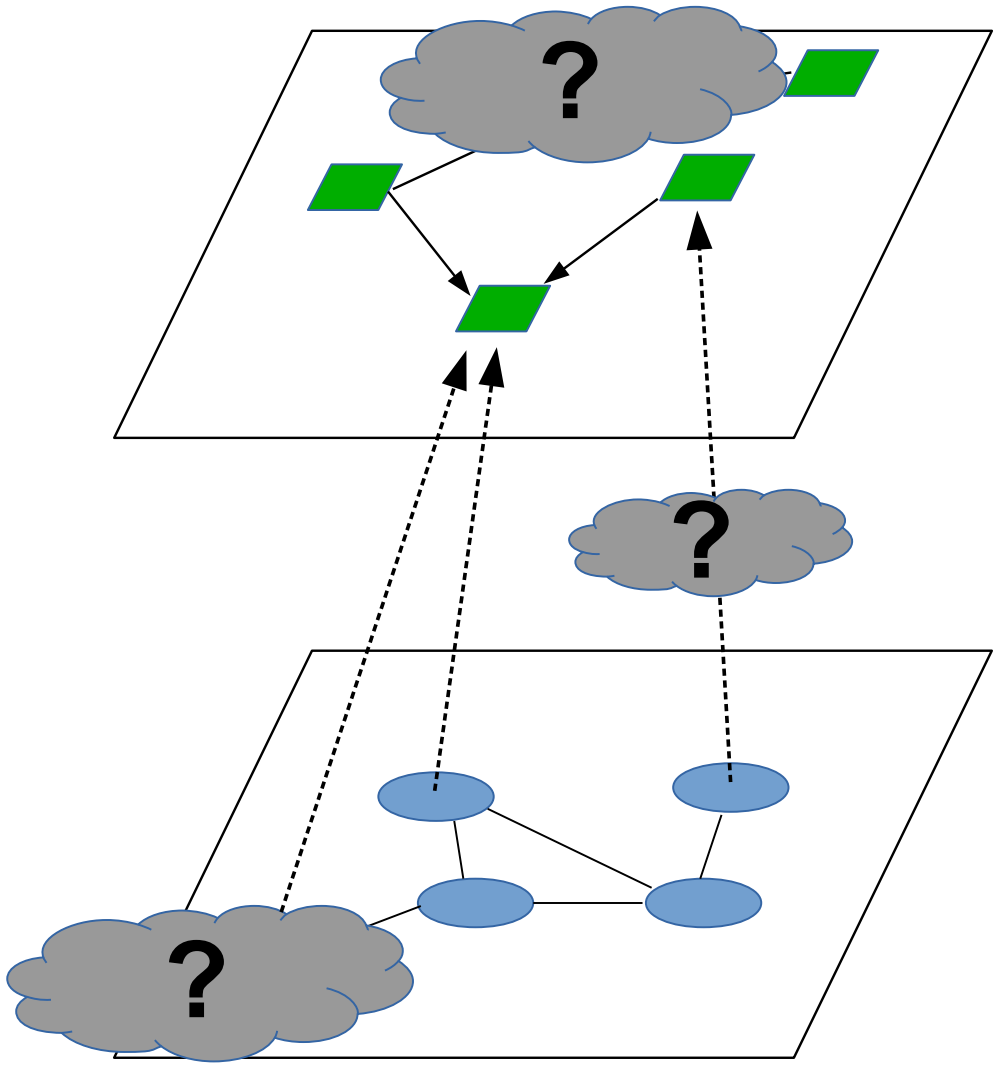
Organ

Cell

Protein

Gene

Sequence



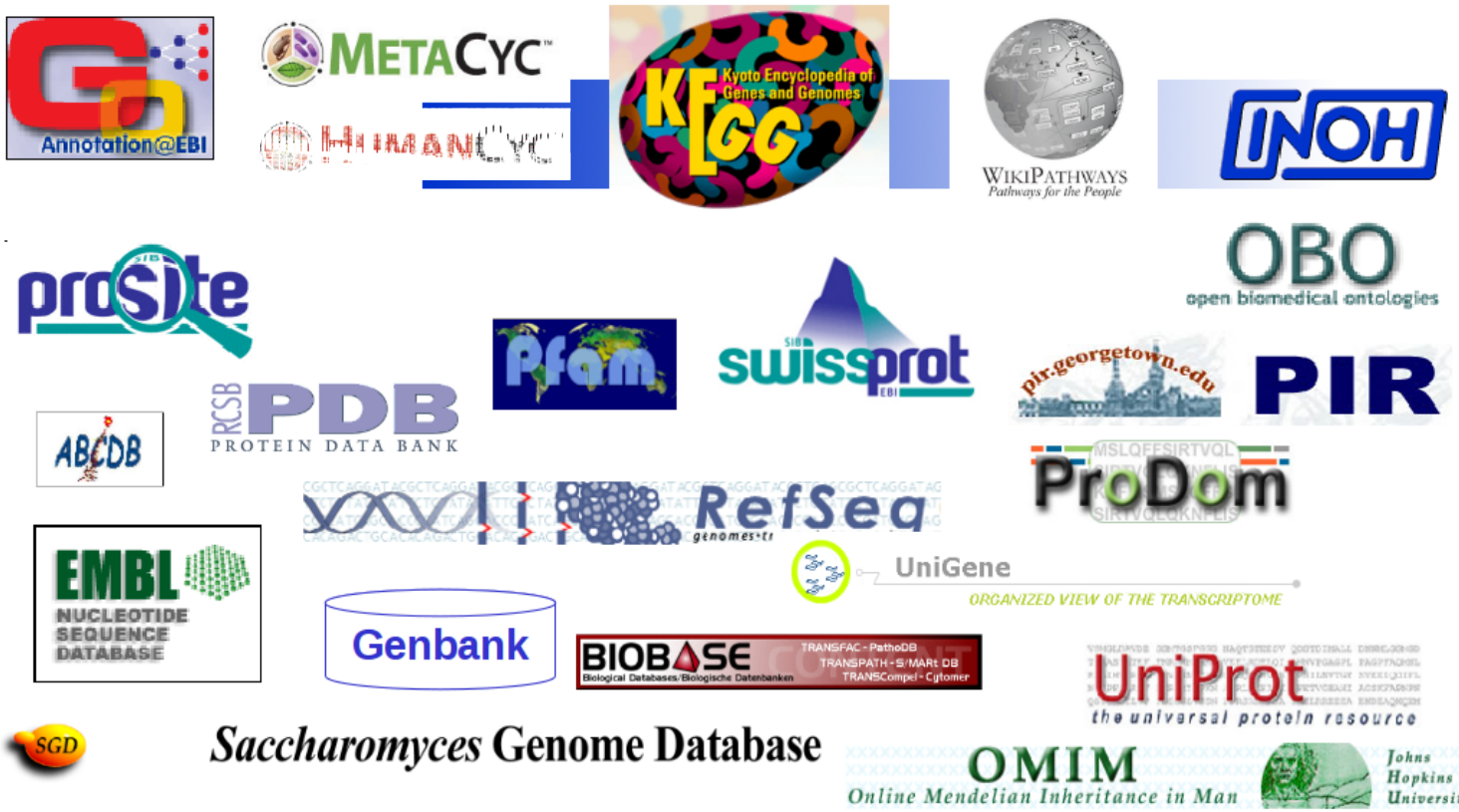
Complexity of life science data : (6) (fast) evolution

- items are added or modified
- items are deprecated

- cascade of dependencies requires to re-run all the experiments that depend on the modified element
  - directly
  - indirectly
- ... by transitivity all the experiments that depend on the results of the previous experiments

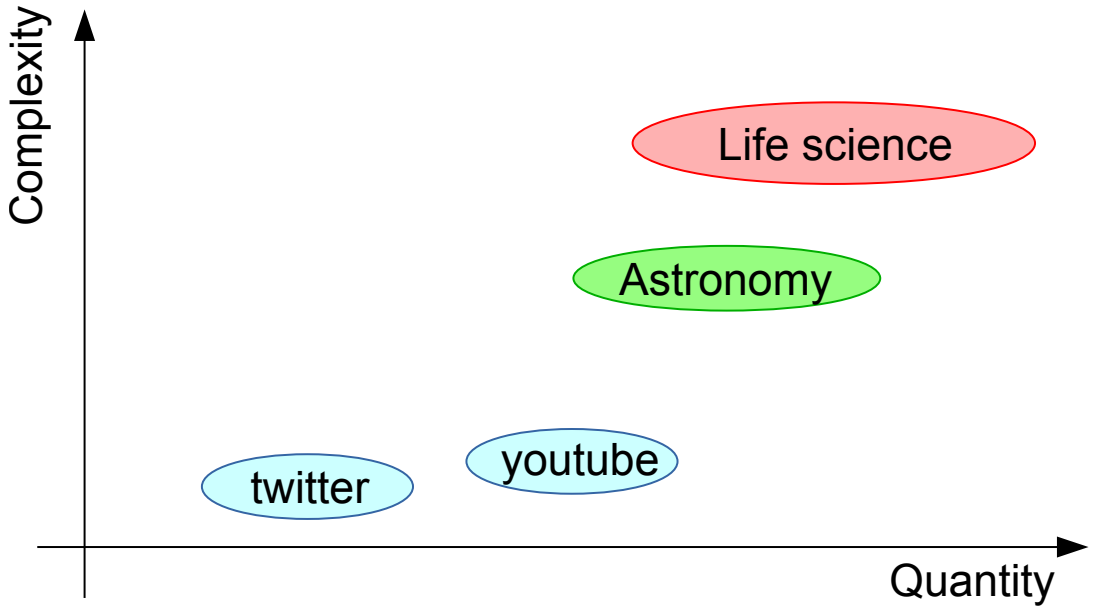
Complexity of life science data : (7) distributed

- 1500+ biological databases [Galperin2015]
- Lack of interoperability
- Some efforts of unified access (BioMart, InterMine...)



Degrees of data complexity

- multiple scales (heterogeneity)
- (highly) interdependent at each scale
- interdependent between scales
- variability
- incompleteness
- evolution
- distributed (and lack of interoperability)



Complexity of life science data

Challenge (computational)

How to handle this complexity ?

- Experts are very good at doing it on their domain (hint)
- The difficulty is to do it systematically
- Expertise = ability to use knowledge for interpreting data
- We should use their expertise, not try to outperform them

Capturing expertise with annotations

Annotation

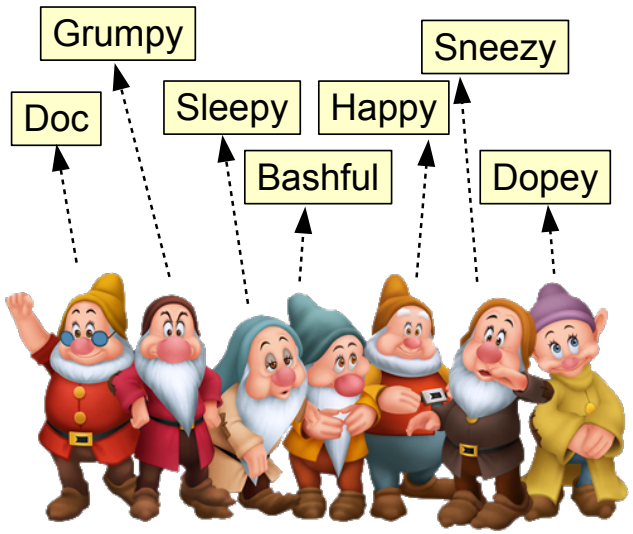
Annotation = result of some interpretation process

Capturing expertise with annotations

Annotation

Annotation = result of some interpretation process

- ideally by an expert (from big data to smart data)
- usually requires some background knowledge
- formalisation ranging from free text to controlled vocabularies to (shared) semantic framework [semantic spectrum]



Using annotations for overcoming data complexity

Add annotations ? But we have too much data already !

Benefits

- can be used as proxy to complex data
- simplifies by providing a compact abstraction
- overcomes variability
- enriches by making explicit the underlying meaning

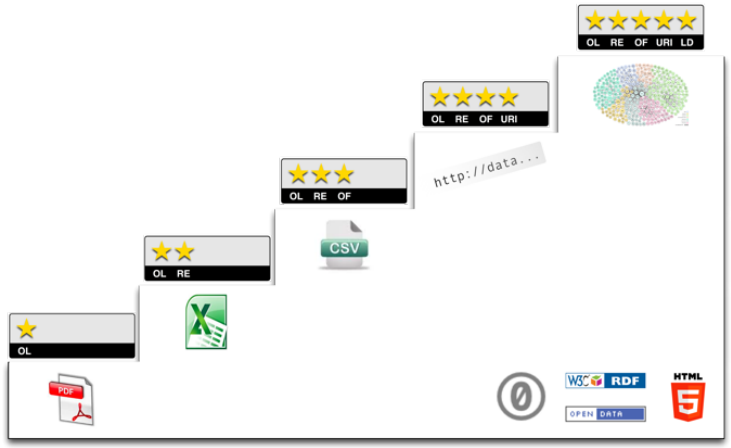
Storing, sharing and reusing these annotations is the key to life science data systematic analysis

Linked data for representing and combining annotations

Relying on annotations and symbolic knowledge is not specific to life sciences

W3C : from the Web of documents to the Web of data

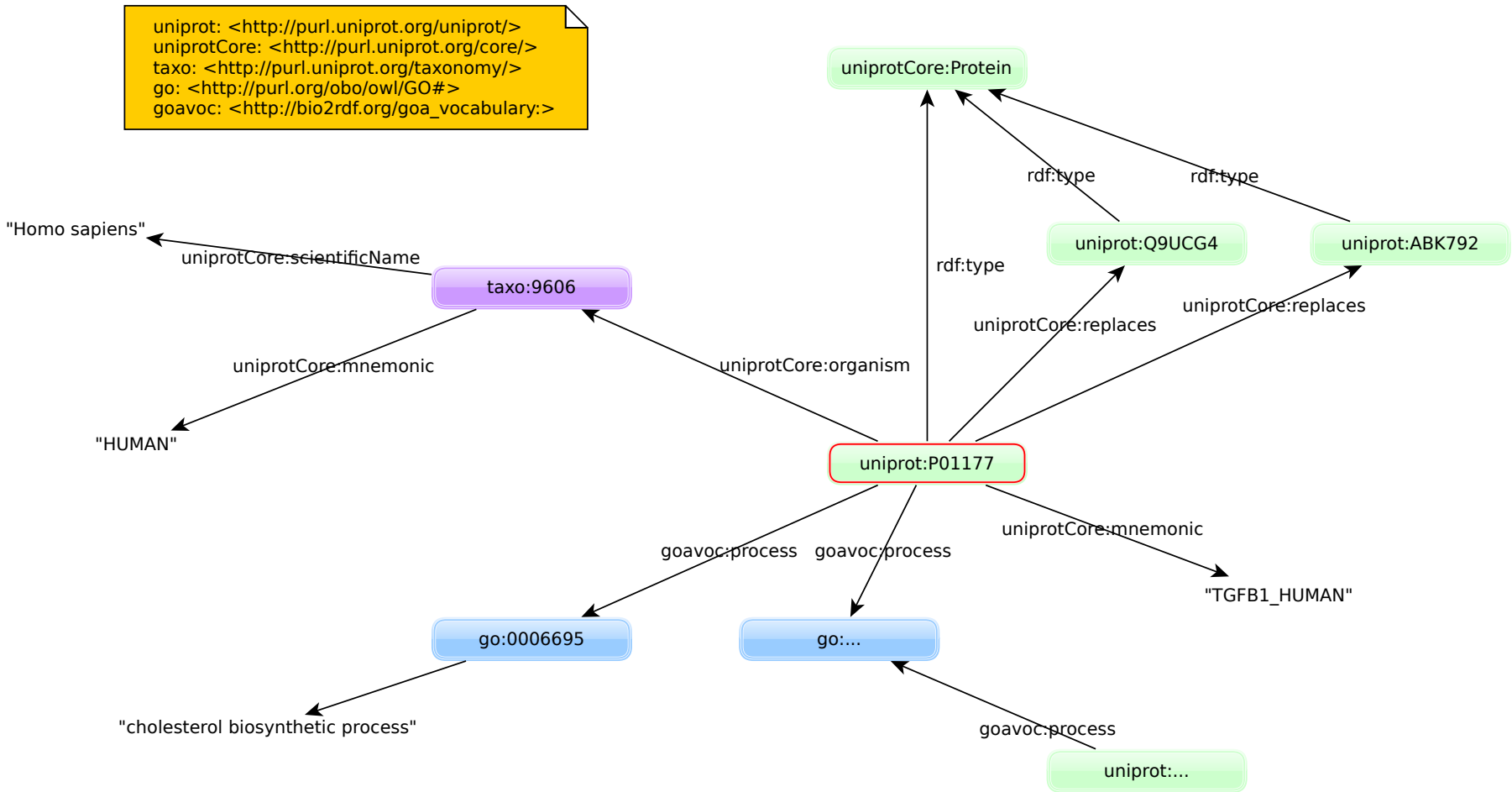
- distributed
- interoperable
- combinable
- compatible with automatic processing including reasoning



http ://5stardata.info

(Simplified) annotations for TGF-β1 (uniprot :P01177)

Annotations are represented as (typed) relations between entities



Ontologies

Knowledge underlying annotations remains to be represented

- "Much of biology works by applying prior knowledge [...] to an unknown entity" [Stevens2000]
- "The complex biological data stored in bioinformatics databases often require the addition of knowledge to specify and constrain the values held in that database" [Stevens2000]

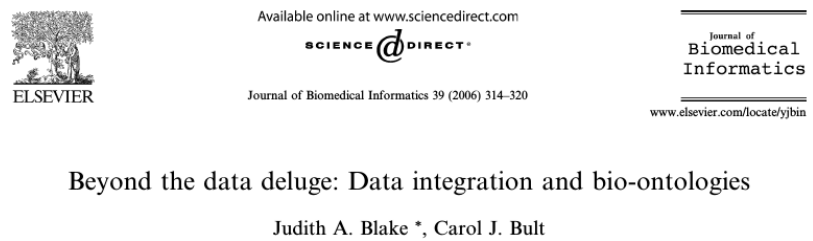
Ontology

Formal representation of knowledge in which the essential terms are combined with structuring rules that describe the relationships between them [Bard2004]


Brief Bioinform. 2000 Nov;1(4):398-414.

Ontology-based knowledge representation for bioinformatics.

Stevens R[1], Goble CA, Bechhofer S.


ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT

Journal of Biomedical Informatics 39 (2006) 314–320

Journal of Biomedical Informatics

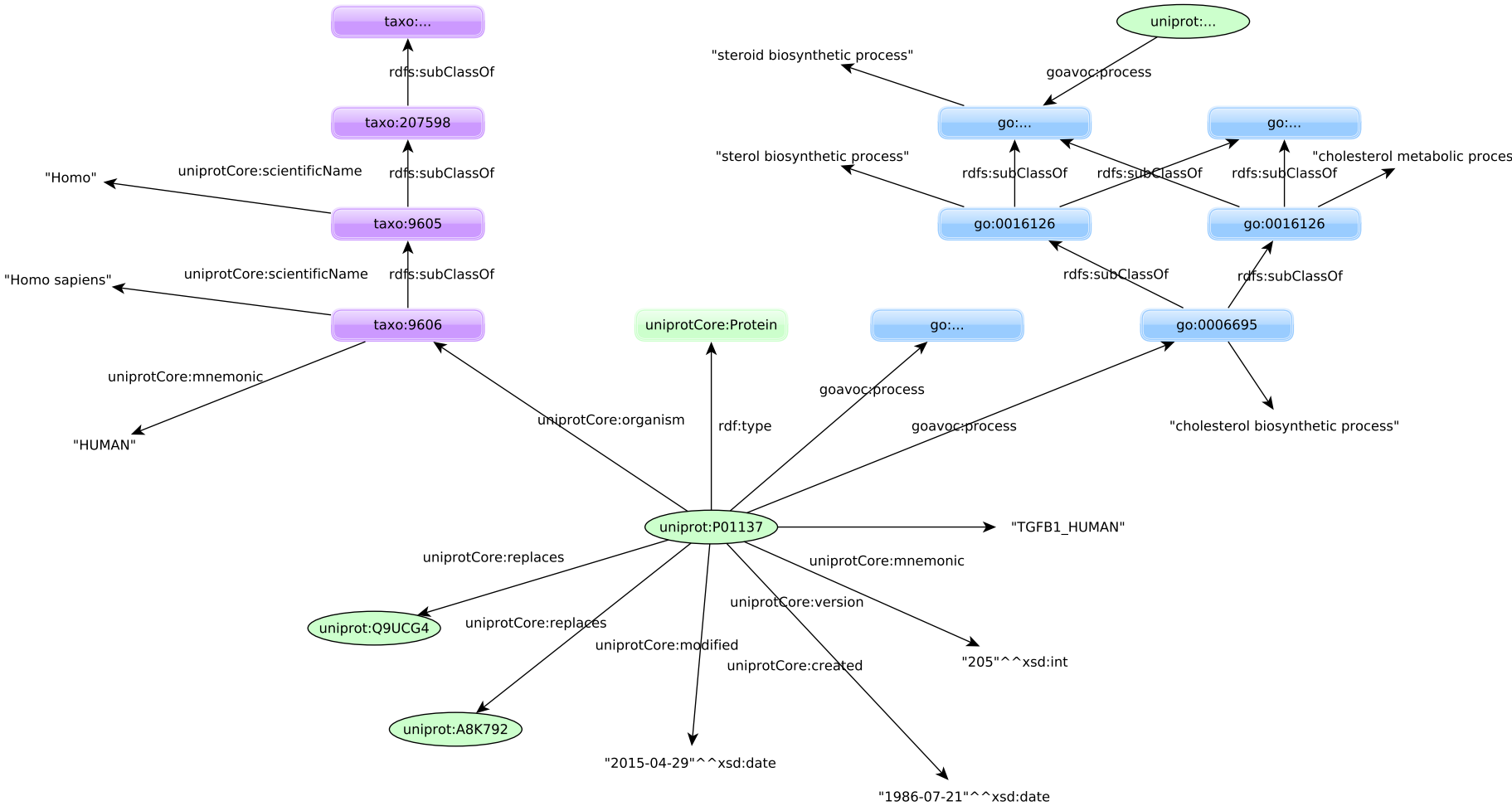
www.elsevier.com/locate/yjbin

Beyond the data deluge: Data integration and bio-ontologies

Judith A. Blake *, Carol J. Bult

Ontologies specify the meaning of annotations

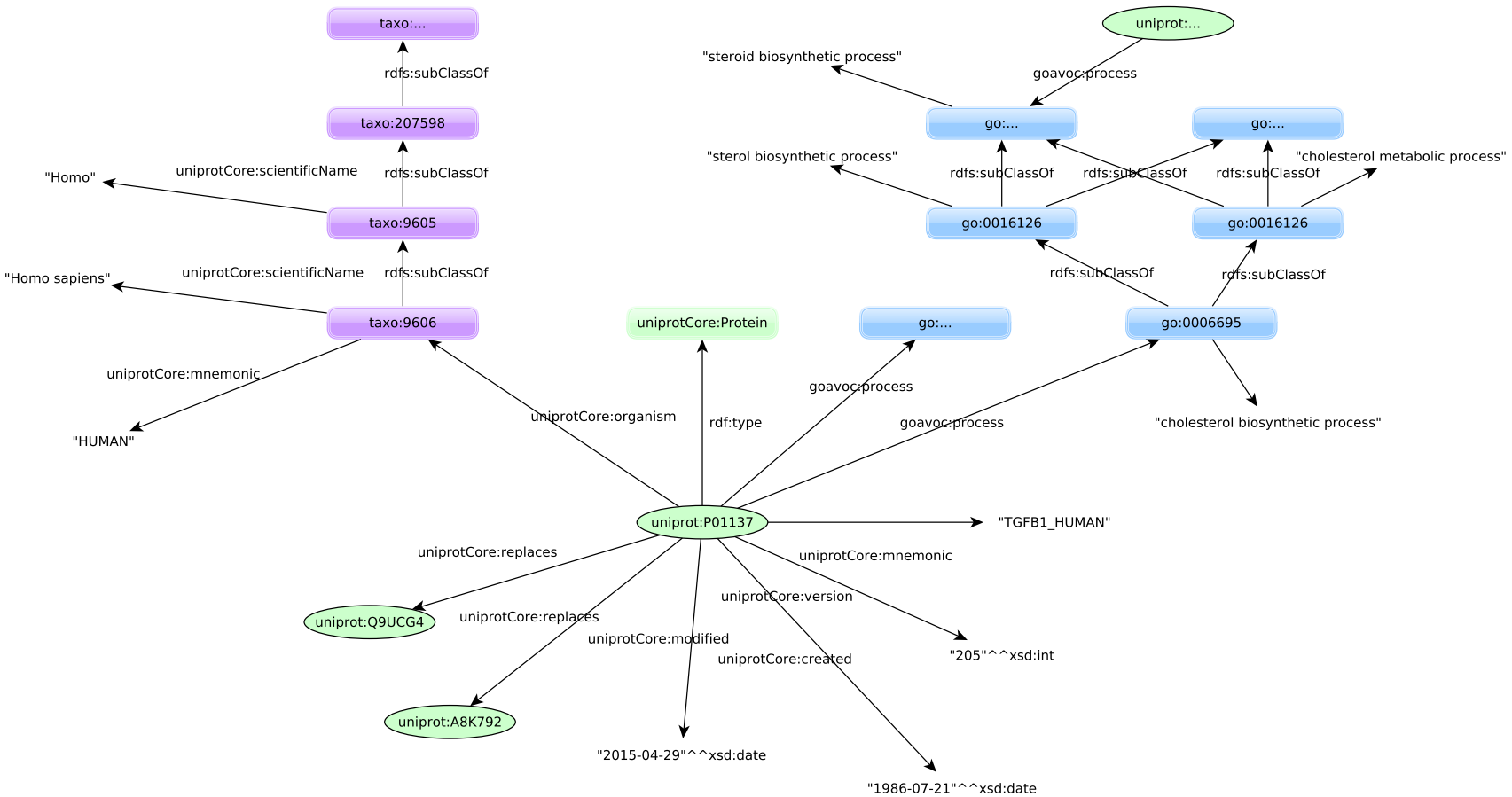
Knowledge is represented as relations between sets of entities



Ontologies support reasoning about annotations

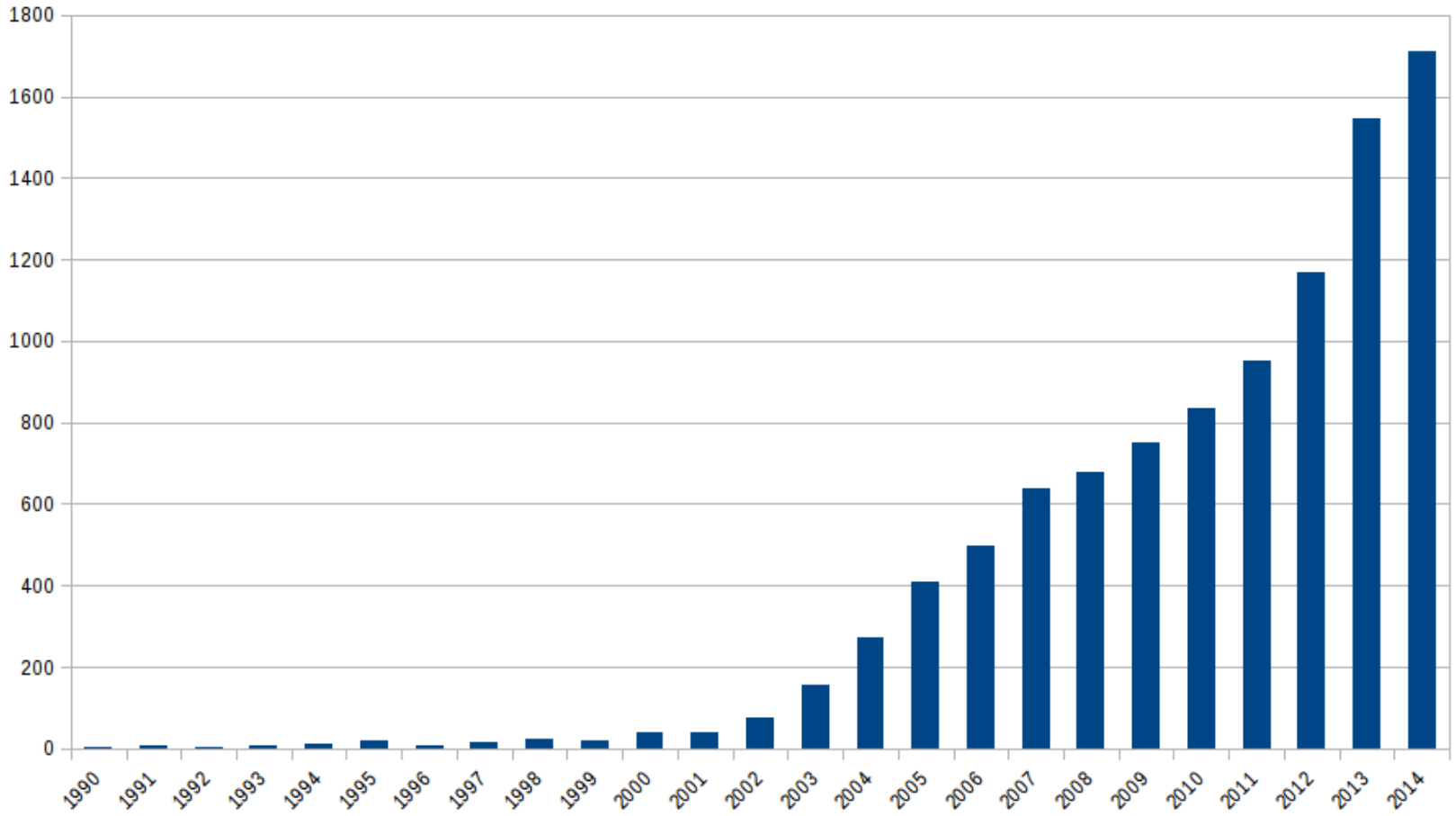
Reasoning

Method for traversing or enriching the graph of data



The ontology deluge (this is a good news !)

Number of PubMed articles mentionning “ontology”



Semantic Web and Linked (Open) Data

Semantic Web offers a unified framework to Linked Data

- **RDF** for representing and aggregating entities descriptions
- **RDFS+OWL** for representing domain knowledge (and combine it with data descriptions)
- **SPARQL** for querying everything (possibly from multiple repositories)

SPARQL endpoints offer unified query access to RDF repositories
ex : Fuseki, Virtuoso,...

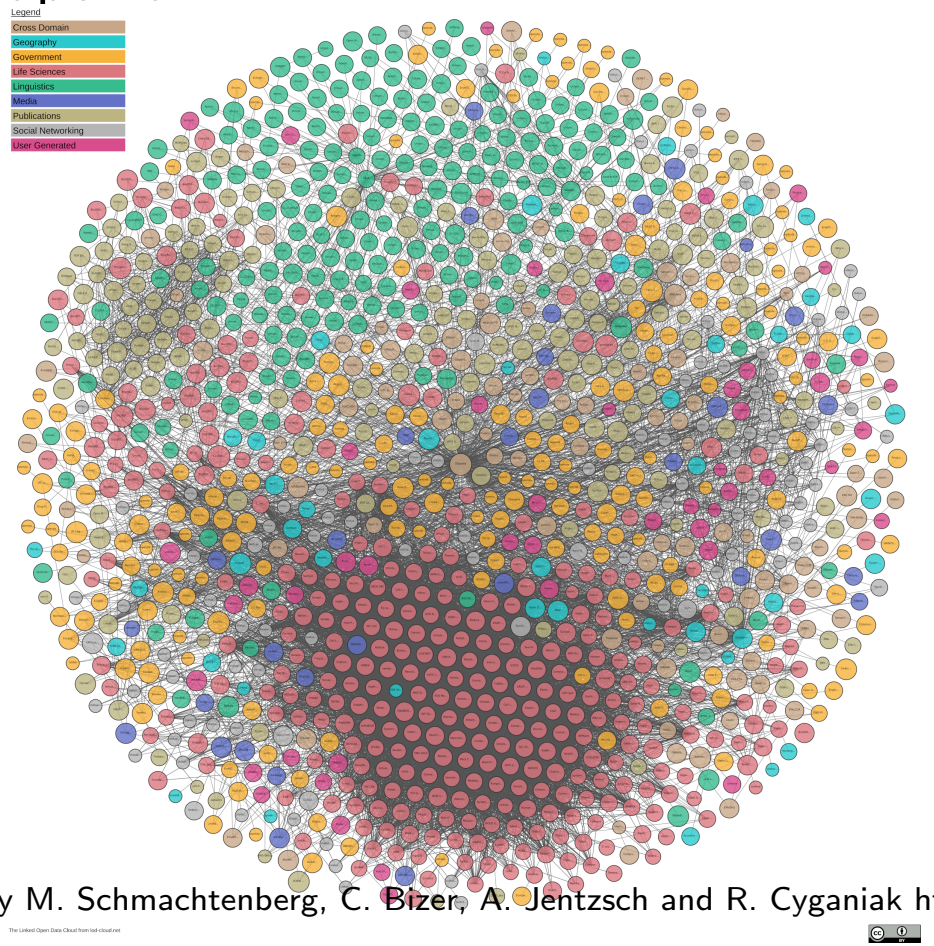
Linked Open Data : a federation of RDF repositories

LODStats (`http://lodstats.aksw.org/`) [Ermilov2016]

- 9960 datasets ; 149.10^9 triples
- general scope ; Life sciences = major field (size+density)

Linked open data (in 2019-03-29)

- RDF repositories can be queried in SPARQL via endpoints
- data from one endpoint can make references to data from another endpoint



Linked open data cloud, by M. Schmachtenberg, C. Bizer, A. Jentzsch and R. Cyganiak http ://lod-cloud.net/

Semantic Web

- general framework relevant for life sciences
- widely adopted by data scientists
- instrumental for future scientific breakthrough

Adoption challenge : Linked data are here... but still have to be adopted by end users

"Real" users

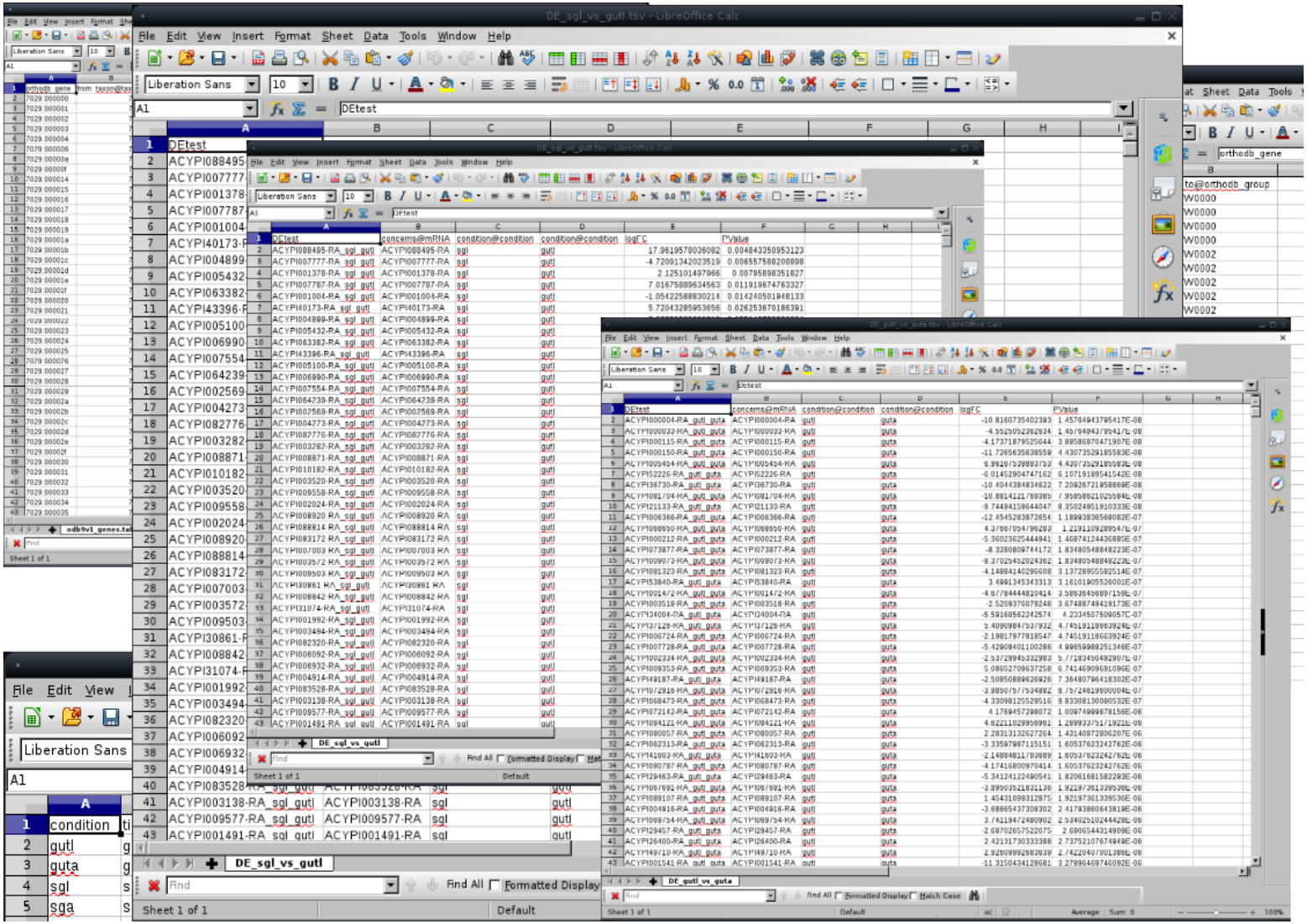
- do not contribute (yet) their data to the LOD cloud
- do not use the LOD cloud for analyzing their own data (yet)

IT challenges

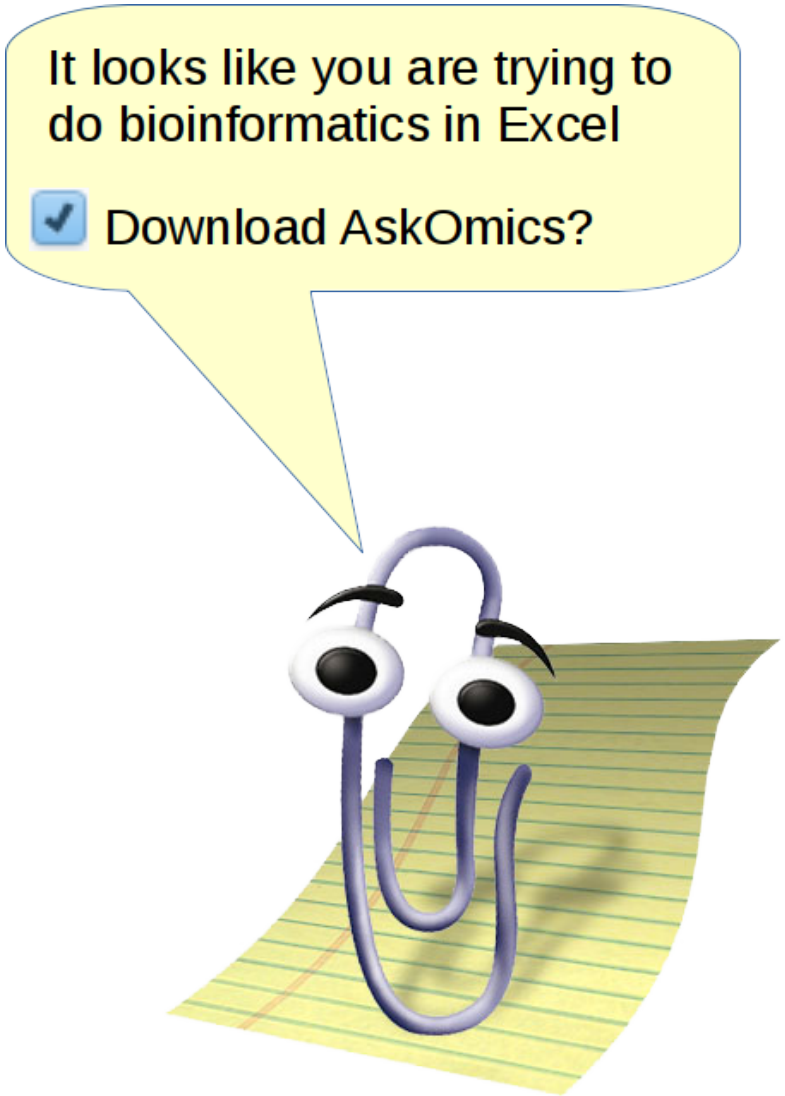
- complex and semantically-rich queries
- over multiple datasets
- containing complex data
- with acceptable response time

Moving individual life science projects to the Semantic Web

End users project's data (aka death by spreadsheet)



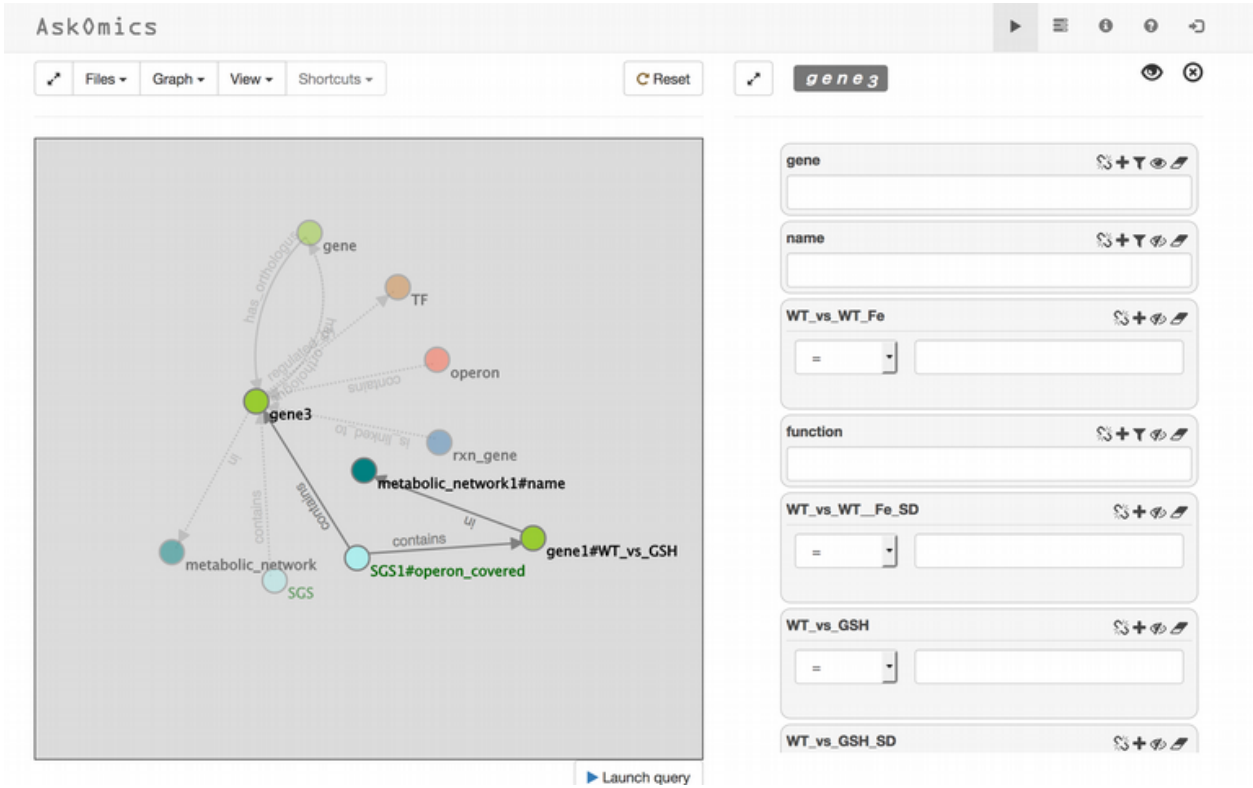
Death by spreadsheet : the worst is yet to come !


It looks like you are trying to do bioinformatics in Excel
Download AskOmics?

AskOmics : bridge btw domain experts and Semantic Web

AskOmics is usefull for :

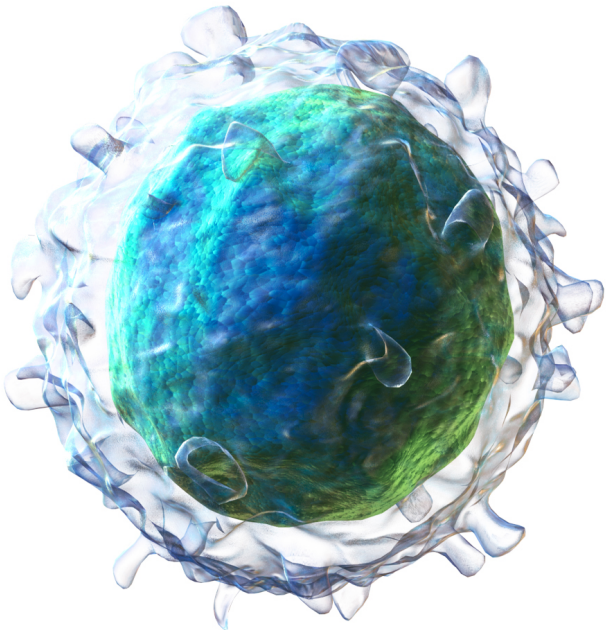
- Integrating data
- Querying data



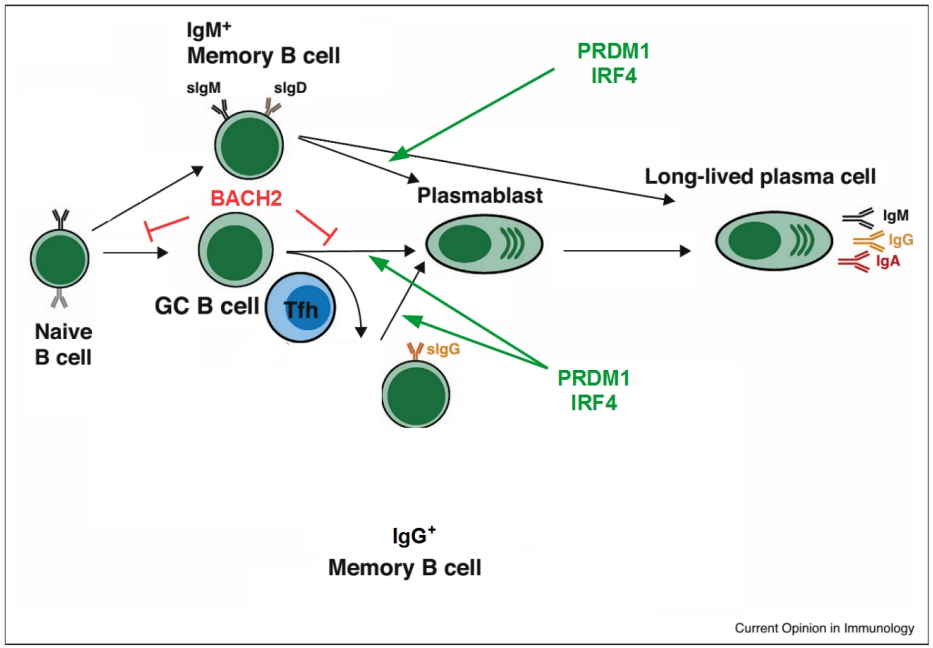
Identifying regulators for B cells differentiation

Collaboration with F. Chatonnet and T. Fest
(INSERM U917 MicMac, CHU Rennes)

- Marine Louarn's M2 internship (January–June 2017)
- INSERM-INRIA PhD since October 2017



Context : Lymphocyte differentiation



NBC differentiation *[Phan]*

- B cells differentiation into plasma cells : immune response.
- Memory B cells : faster differentiation, vaccine principle.
- Can we find the regulation candidates ?

We are looking for new genetic and epigenetic regulation candidates

Gene regulation

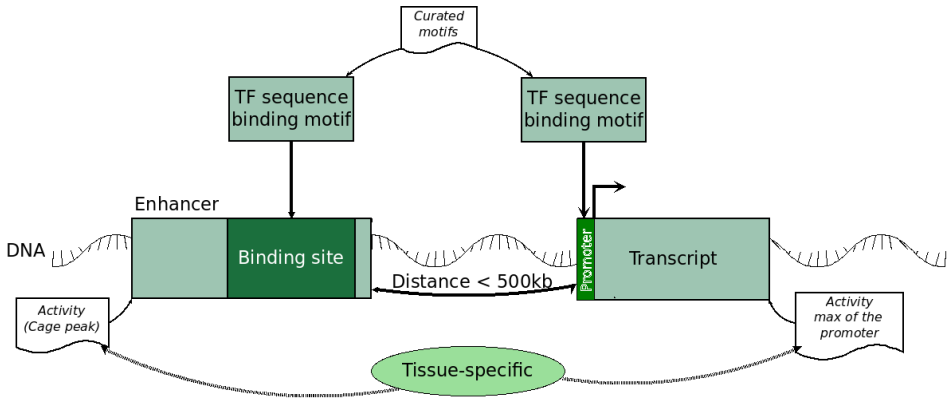
better understanding of

- cell differentiation
- cell identity
- cell function, adaptation and transformation

mediated by Transcription Factors that bind to either

- promoters
- enhancers

only works if the TF's binding site is in open 3D conformation



Gene regulatory networks

typed (induction or inhibition) relations btw a TF and a gene

- ENCODE
- FANTOM5
- RoadMap Epigenomics

- low compliance with FAIR guidelines
- reuse is difficult

The Regulatory Circuits project case study

`http://regulatorycircuits.org`

Nat Methods. 2016 Apr;13(4):366-70. doi: 10.1038/nmeth.3799. Epub 2016 Mar 7.

Tissue-specific regulatory circuits reveal variable modular perturbations across complex diseases.

Marbach D[1,2], Lamparter D[1,2], Quon G[3,4], Kellis M[3,4], Kutalik Z[2,5], Bergmann S[1,2].

Data

- heterogeneous and multi-layers "omics" data
- human patients cells from 394 tissues
- 59 files (6.6GB)

Output

- family of scored tissue-specific regulatory interaction networks
- in text files

The Regulatory Circuits project case study

Method

- incomplete description in supplementary materials
- scripts and algorithms limited to the considered datasets

Limitations

- reproducibility of results
- maintenance/extension with new/additional data sources
- reuse of results for other studies

Objectives

Can the Regulatory Circuits data and workflow be modeled using Semantic Web technologies ?

- identify the relevant files
- propose an RDF data structure
- populate a SPARQL endpoint
- represent the workflow as SPARQL queries

Regulatory Circuits dataset

- 14 input files
- 7 pre-processed intermediary files
- calculated networks (not used in this study)
  - 394 tissue-specific networks
  - 32 high level networks + 40 public networks

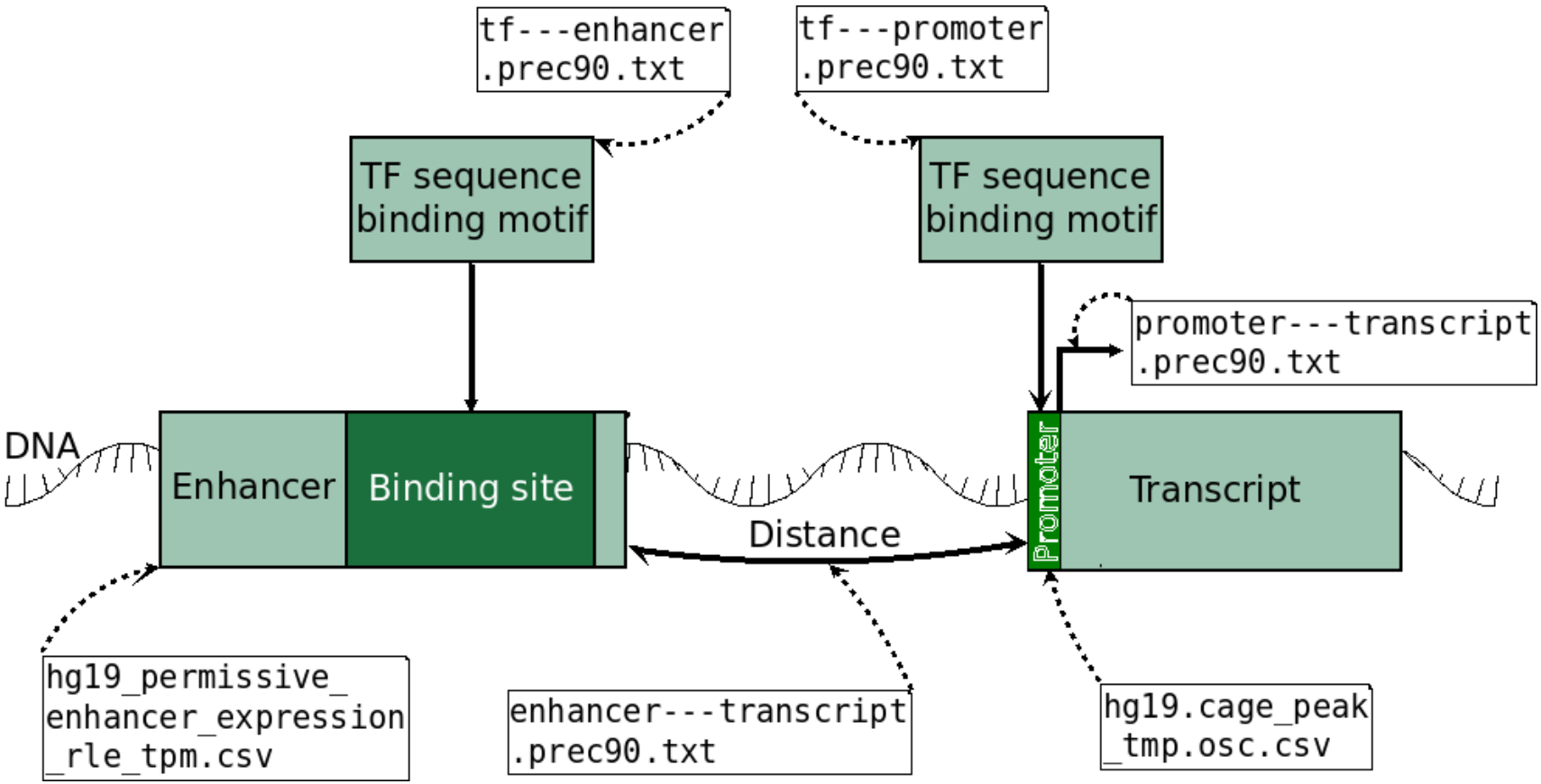
TSV text files

- from 184 to 124.358.159 lines
- 3 to 890 columns
- sometimes with headers (0, 1 or 3 lines)
- sometimes with comments (0, 893 or 1772 lines)

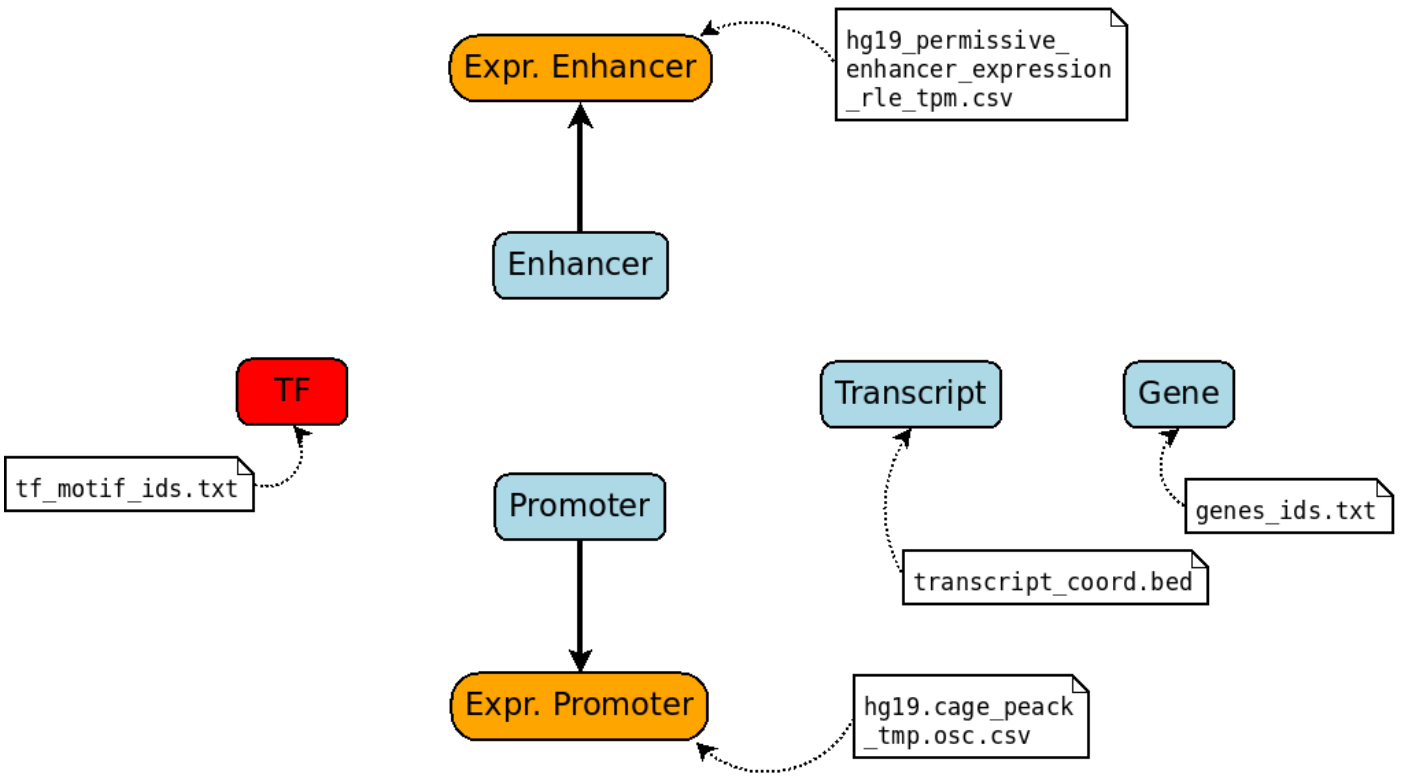
Difficult :

- Determine what is in each file and how are they related ?
- 3 files were mis-formated (offset between columns and header)

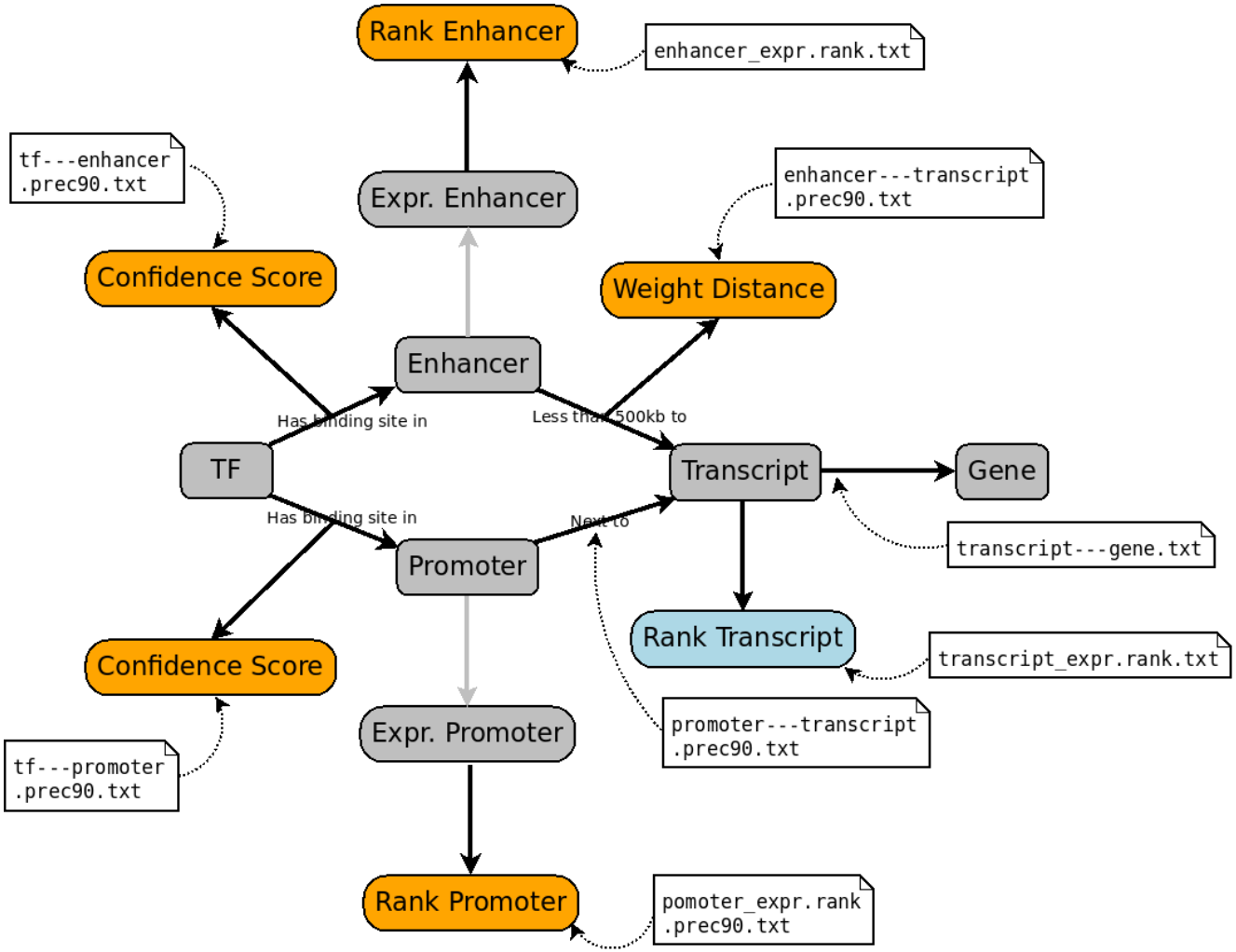
Biological background helped to infer the relations between the data files



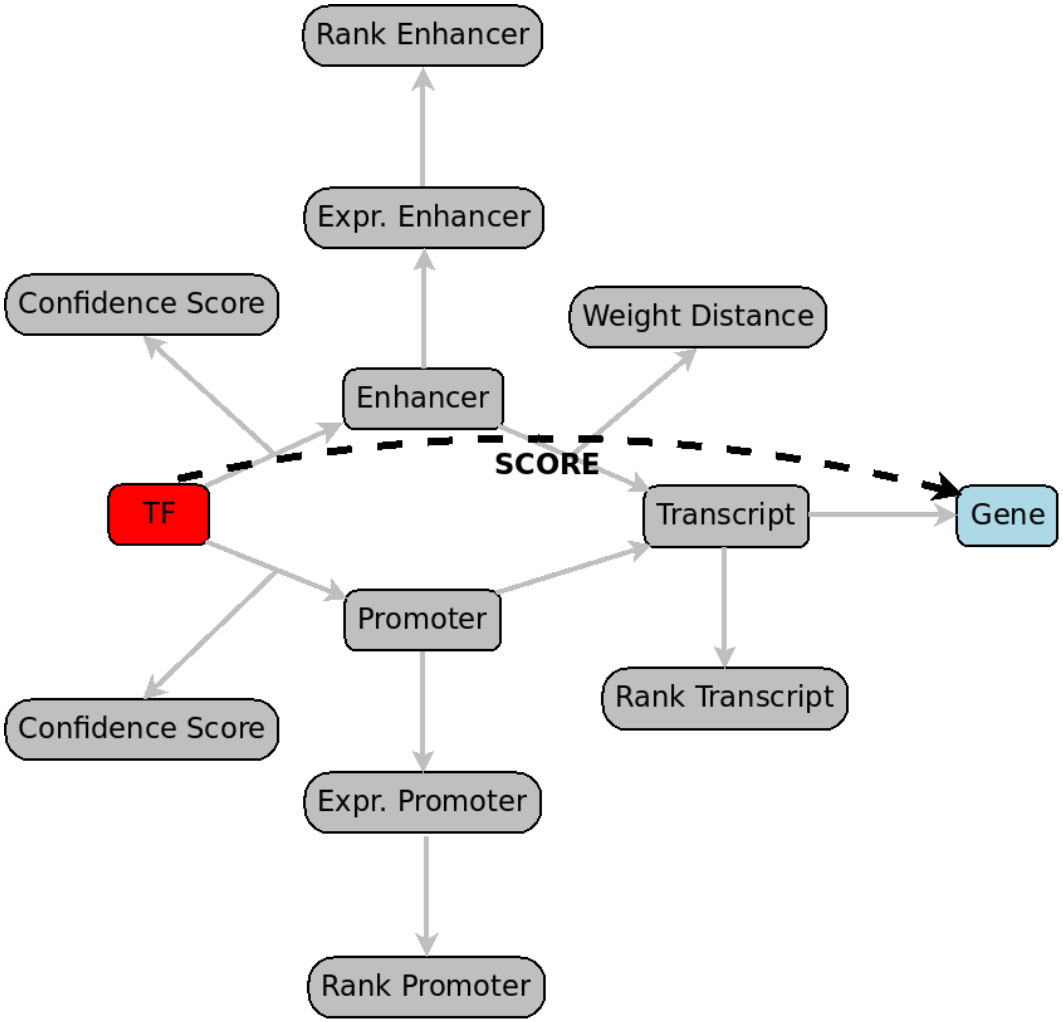
Entities


Expr. Enhancer
hg19_permissive_
enhancer_expression
_rle_tpm.csv
Enhancer
TF
tf_motif_ids.txt
Transcript
Gene
Promoter
genes_ids.txt
transcript_coord.bed
Expr. Promoter
hg19.cage_peack
_tmp.osc.csv

Add relations from intermediary files

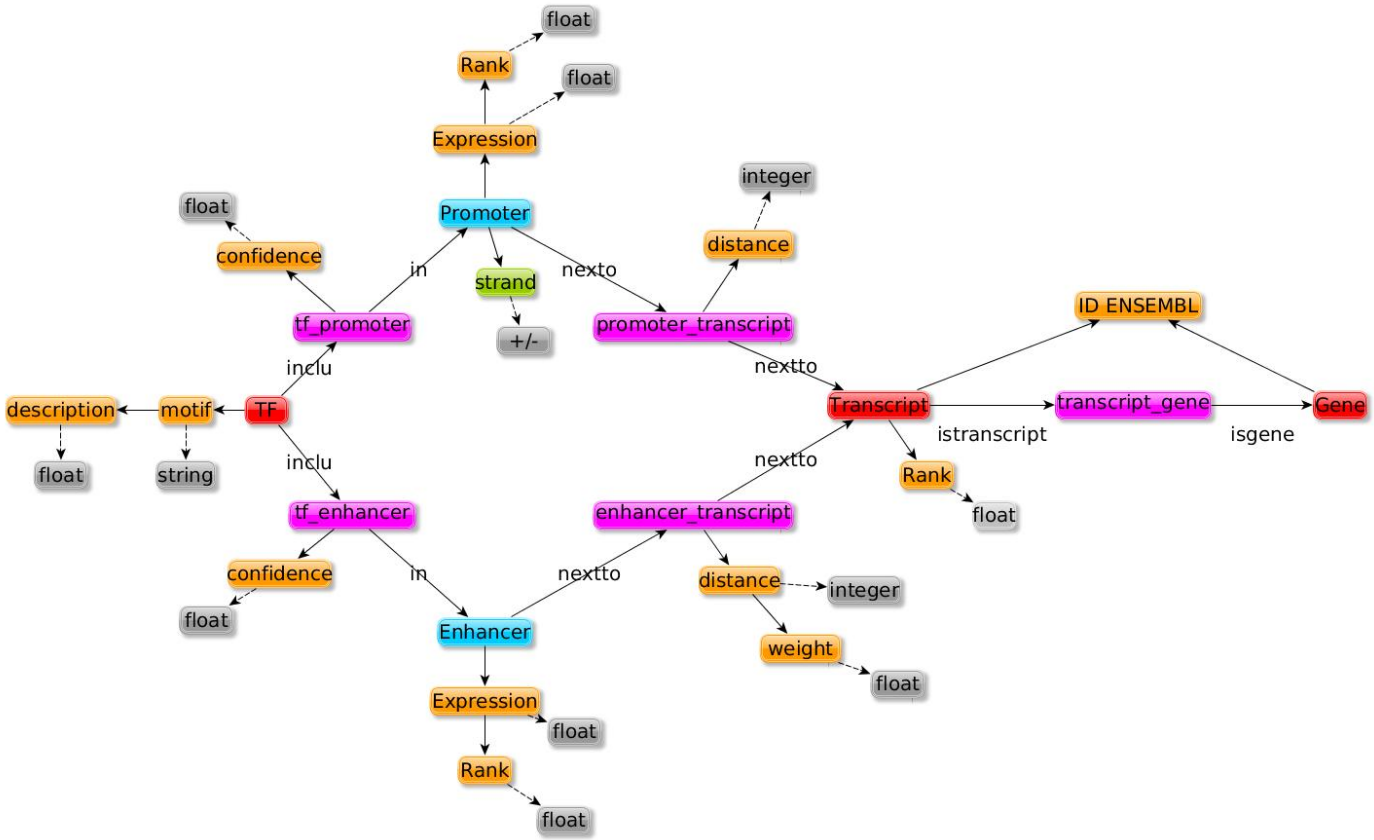


RC workflow for infering TF-genes relations

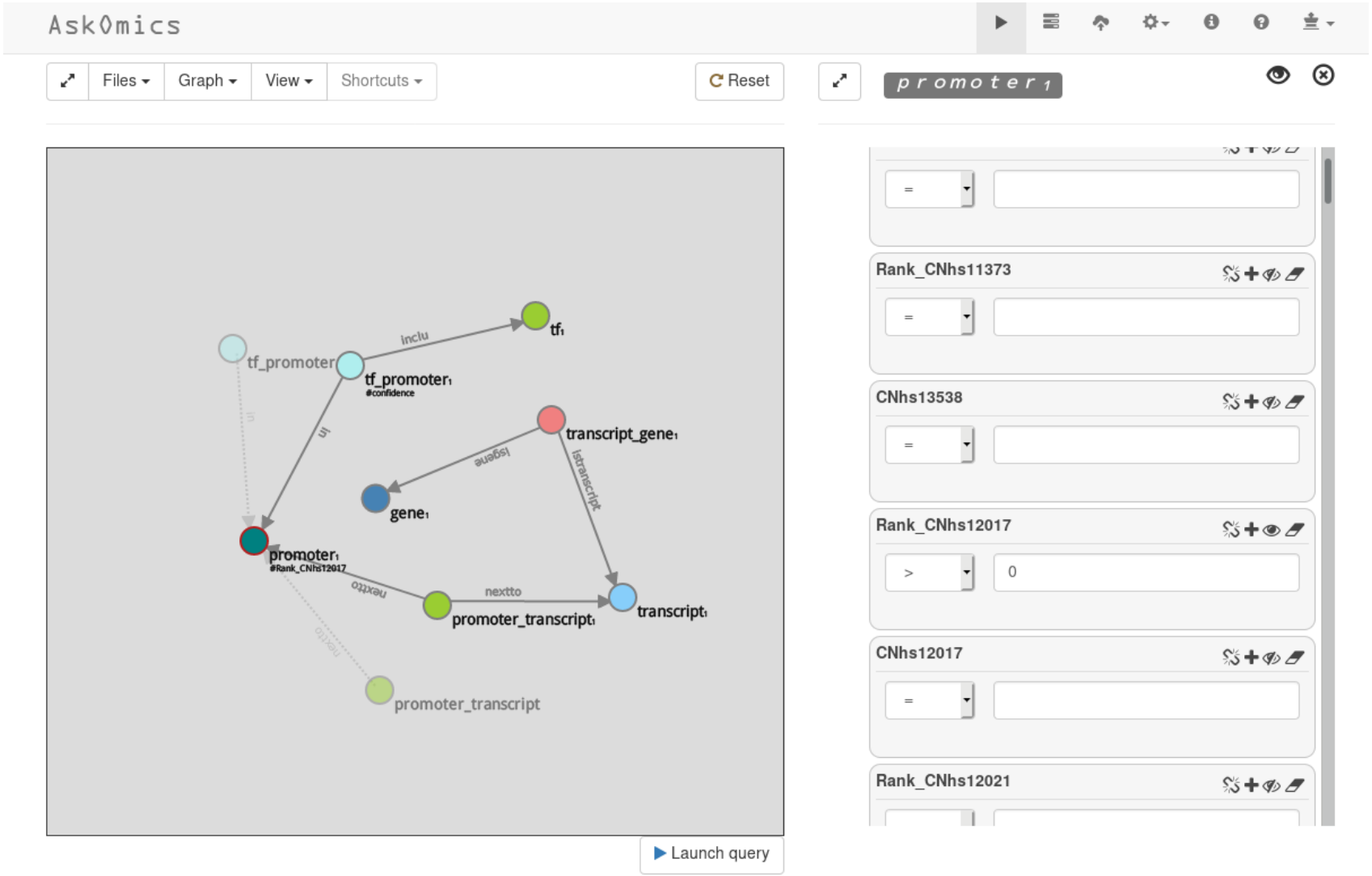


RDF data structure

- 3.226.341 entities
- 335.429.988 triples
- `https://regulatorycircuits-rdf.genouest.org/sparql/`



TF-gene relations through promoters (without score)



TF-gene relations through promoters (without score)

```
SELECT DISTINCT ?tf1 ?gene1
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_promoter1 rdf:type user:tf_promoter.
  ?tf_promoter1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?promoter1 rdf:type user:promoter.
  ?promoter1 askomics:Rank_CNhs12017 ?Rank_CNhs12017P.
  FILTER ( ?Rank_CNhs12017P > 0 ).
  ?promoter_transcript1 rdf:type user:promoter_transcript.
  ?transcript1 rdf:type user:transcript.
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_promoter1 askomics:inclu ?tf1.
  ?tf_promoter1 askomics:in ?promoter1.
  ?promoter_transcript1 askomics:nextto ?promoter1.
  ?promoter_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
ORDER BY ?tf1 ?gene1
```

TF-gene relations through promoters (with score)

```
SELECT DISTINCT ?tf1 ?gene1 (max(xsd:float(?confidence1) *
    xsd:float(?confidence1) * xsd:float(?Rank_CNhs12017P) *
    xsd:float(?Rank_CNhs12017P)) AS ?weightP)
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_promoter1 rdf:type user:tf_promoter.
  ?tf_promoter1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?promoter1 rdf:type user:promoter.
  ?promoter1 askomics:Rank_CNhs12017 ?Rank_CNhs12017P.
  FILTER ( ?Rank_CNhs12017P > 0 ).
  ?promoter_transcript1 rdf:type user:promoter_transcript.
  ?transcript1 rdf:type user:transcript.
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_promoter1 askomics:inclu ?tf1.
  ?tf_promoter1 askomics:in ?promoter1.
  ?promoter_transcript1 askomics:nextto ?promoter1.
  ?promoter_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
GROUP BY ?tf1 ?gene1
```

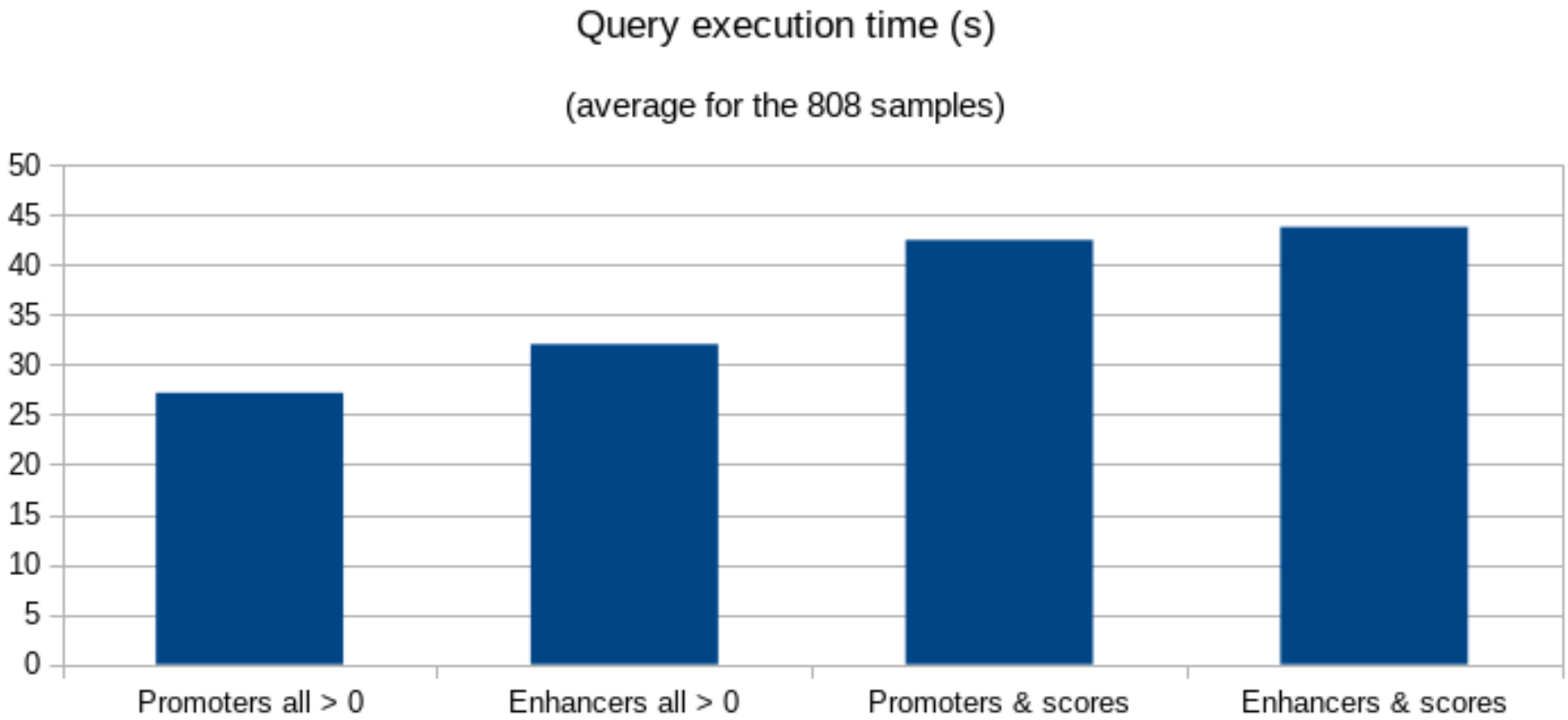
TF-gene relations through enhancers (without score)

```
SELECT DISTINCT ?tf1 ?gene1
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_enhancer1 rdf:type user:tf_enhancer.
  ?tf_enhancer1 askomics:confidence ?confidence1.
 FILTER ( ?confidence1 > 0 ).
  ?enhancer1 rdf:type user:enhancer.
  ?enhancer1 askomics:Rank_CNhs12017 ?Rank_CNhs12017E.
  FILTER ( ?Rank_CNhs12017E > 0 ).
  ?enhancer_transcript1 rdf:type user:enhancer_transcript.
  ?enhancer_transcript1 askomics:weight ?weight1.
  FILTER ( ?weight1 > 0 ).
  ?transcript1 rdf:type user:transcript.
  ?transcript1 askomics:CNhs12017 ?CNhs12017T.
  FILTER ( ?CNhs12017T > 0 ).
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_enhancer1 askomics:inclu ?tf1.
  ?tf_enhancer1 askomics:in ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
ORDER BY ?tf1 ?gene1
```

TF-gene relations through enhancers (with score)

```
SELECT DISTINCT ?tf1  ?gene1 (max(xsd:float(?confidence1) *
  xsd:float(?confidence1) * xsd:float(?weight1)*
  xsd:float(?weight1) * xsd:float(?CNhs12017T) *
  xsd:float(?Rank_CNhs12017E) ) AS ?weightE)
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_enhancer1 rdf:type user:tf_enhancer.
  ?tf_enhancer1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?enhancer1 rdf:type user:enhancer.
  ?enhancer1 askomics:Rank_CNhs12017 ?Rank_CNhs12017E.
  FILTER ( ?Rank_CNhs12017E > 0 ).
  ?enhancer_transcript1 rdf:type user:enhancer_transcript.
  ?enhancer_transcript1 askomics:weight ?weight1.
  FILTER ( ?weight1 > 0 ).
  ?transcript1 rdf:type user:transcript.
  ?transcript1 askomics:CNhs12017 ?CNhs12017T.
  FILTER ( ?CNhs12017T > 0 ).
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_enhancer1 askomics:inclu ?tf1.
  ?tf_enhancer1 askomics:in ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
GROUP BY ?tf1 ?gene1
ORDER BY ?tf1 ?gene1
```

Performances



We replaced their whole workflow by 2 SPARQL queries

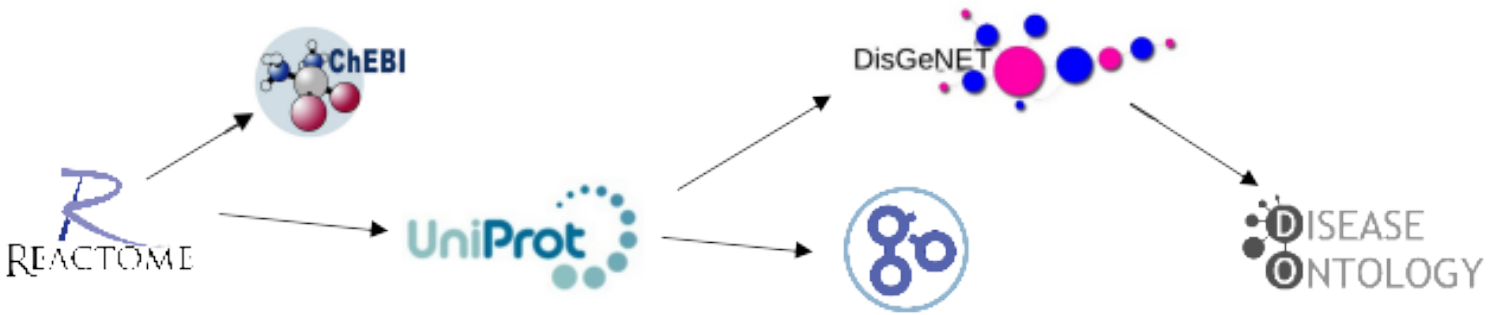
Semantic Web technologies are extensively used for supporting knowledge base interoperability and reusability

Semantic Web technologies are also relevant for original studies

- improve results reproducibility
- improve results updates
- improve results reuse in other studies

Improve federated query processing

Improve federated queries processing

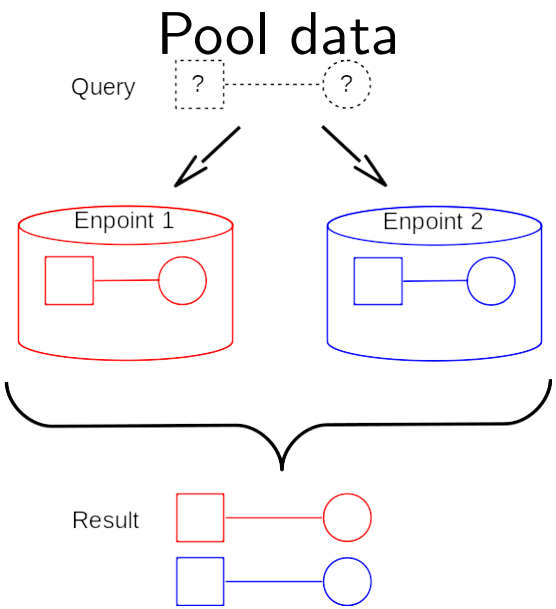


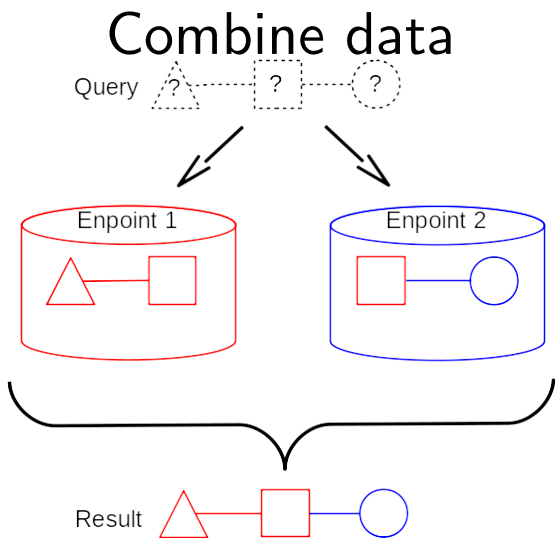
Challenge

Poor performances recognized as a major bottleneck [Bairoch2016]

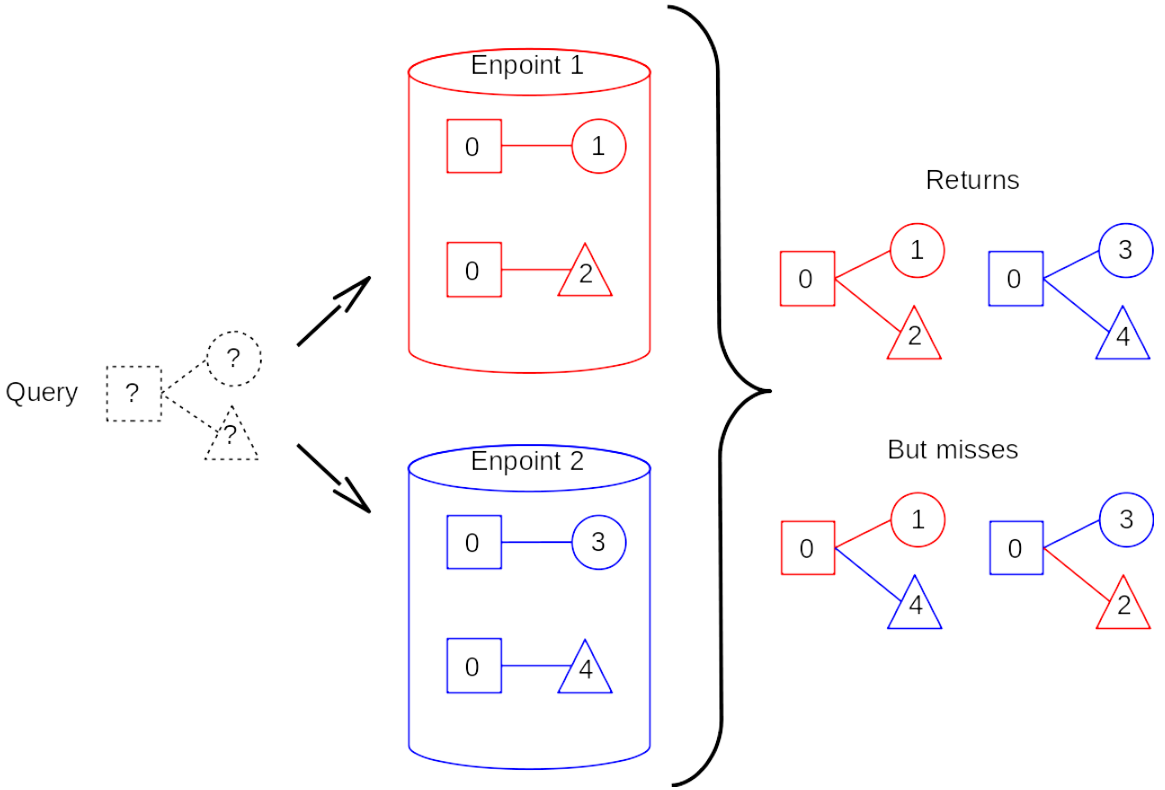
Federated queries principle

- Linked data
  - RDF repositories can be queried in SPARQL via endpoints
  - $\text{data}_{endpoint1}$ can make references to $\text{data}_{endpoint2}$
- Federated queries span several endpoints
  - SPARQL engine propagates the query and merges the results
  - good news : supported by SPARQL language + query engines
  - not so good news : performances :-(



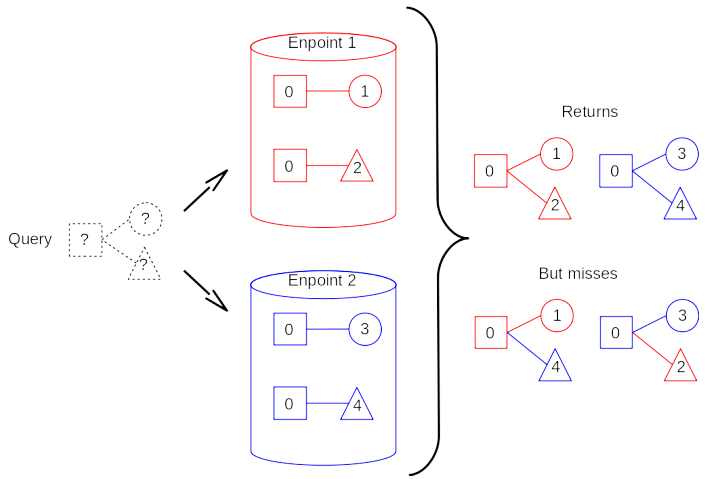


Federated queries difficulty : endpoints not independent



Treating the endpoints independently fails when combining data

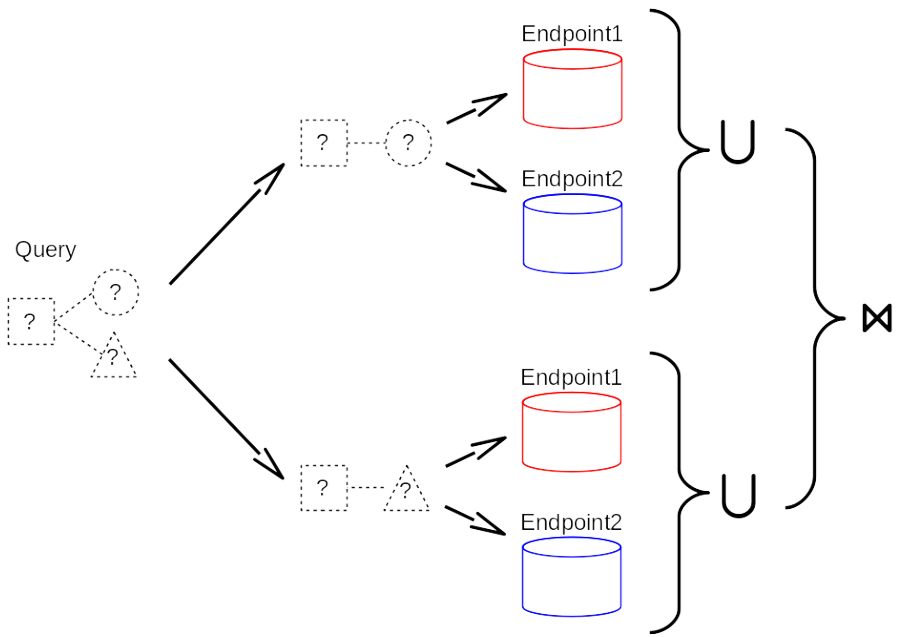
Federated queries difficulty : endpoints can not be merged



Merging the endpoints is not a viable solution either

- each endpoint is potentially big
- merging
  - increases network traffic
  - increases storage consumption
  - decreases query answering performances
  - does not scale up to LOD

Federated queries : q. fragmentation increases complexity



Sending each triple to each endpoint results in

- many subqueries for each endpoint (distant server overload)
- many unions and joins (local engine overload)
- potential transfer of large quantities of data before performing the joins, even if it ultimately few results (network overload)

Processing federated queries : general approach

Decompose the query into fragments

The fewer fragments the better : reduces joins

s. selection for each fragment, select the relevant endpoints

The fewer endpoints the better (but no false negatives !) : reduces joins

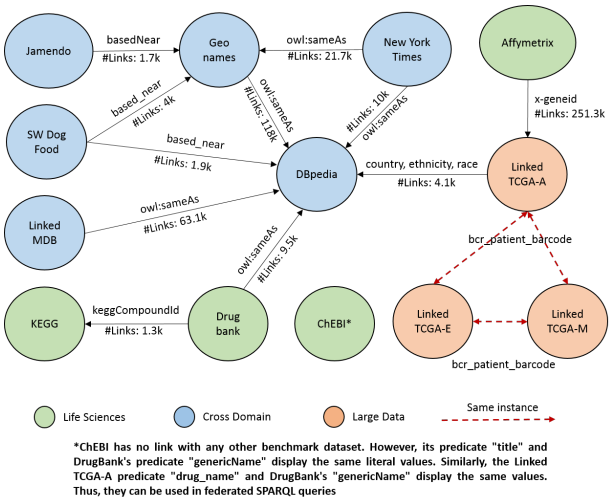
Determine the order for processing the fragments (q. planning)

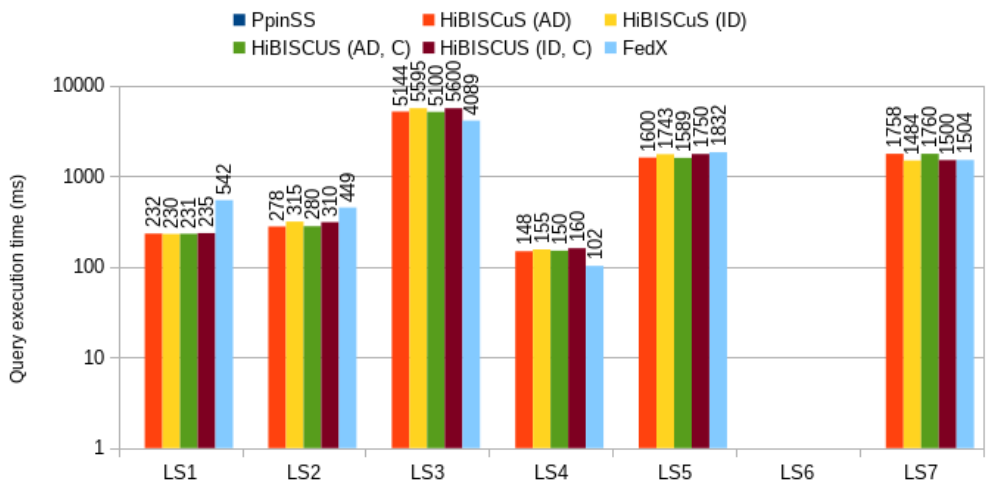
Start by the most selectives, maybe parallelize, and potentially rewrite the subqueries

These three aspects can be inter-dependent

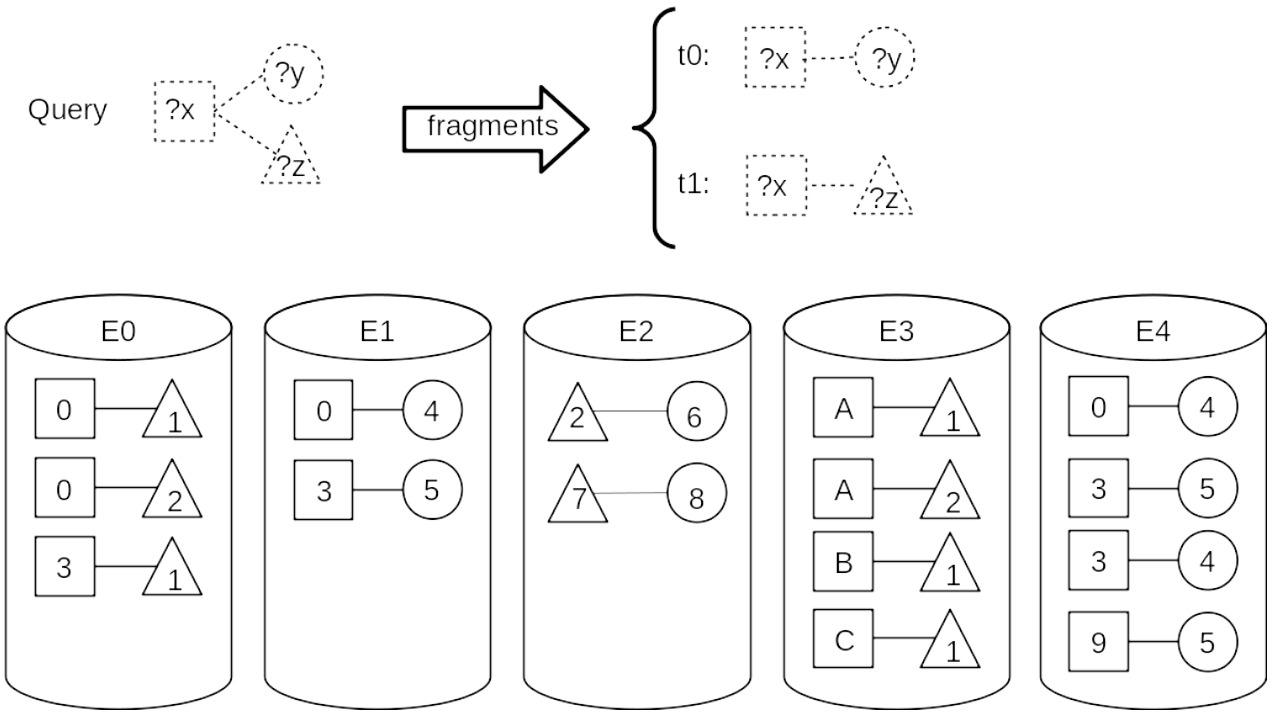
Federated queries : the LargeRDFBench benchmark

- 13 datasets ; $> 10^9$ triples
- 40 queries, including 7 related to Life sciences
- **Both FedX and HiBISCuS timeout for LS6**





Source selection



Naive approach

- `t0` : {E0...E4}
- `t1` : {E0...E4}

8 unions + 1 join

Structure

- `t0` : {E1, E4}
- `t1` : {E0, E3}

2 unions + 1 join

Structure + content

- `t0` : {E1, E4}
- `t1` : {E0}

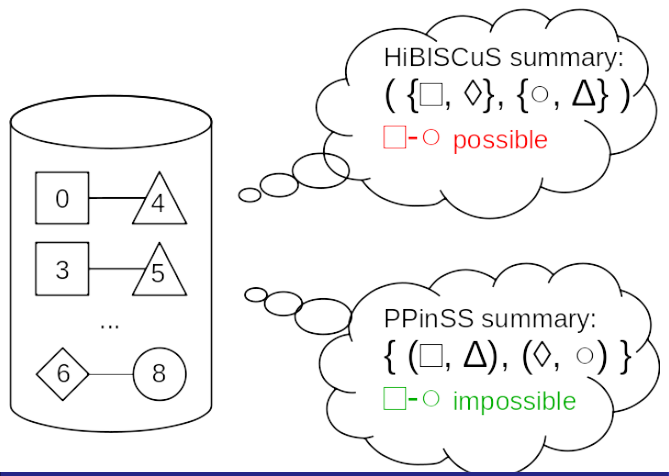
1 union + 1 join

Endpoint summaries in FederatedQueryScaler

Similar to HiBISCuS, our summaries associate the relations with patterns of the subjects and the objects identifiers

Our summaries :

- use richer patterns of identifiers
  - (-) take longer to compute
  - (-) use more memory
  - (+) are more discriminant
- can capture identifiers patterns coupling with sets of pairs of patterns (instead of pairs of sets of patterns)



Evaluation

We compared :

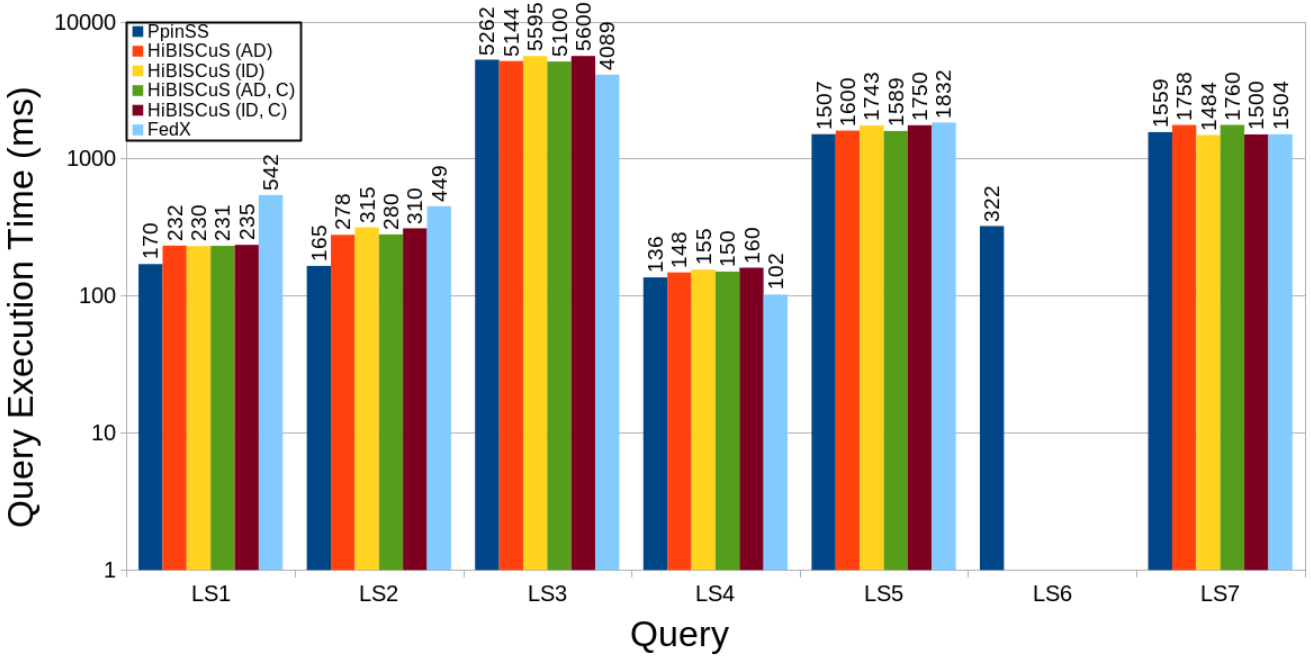
- FedX (no index)
- HiBISCuS (index based on pairs of sets of simple patterns)
- PPinSS : HiBISCuS with our summary-based source selection

We used :

- 13 endpoints (total $> 10^9$ triples)
- the 7 life science queries among the 32 from the LargeRDFBench benchmark

- Our index was larger than HiBISCuS' but remained acceptable (27Mb)
- We selected fewer sources (30) than HiBISCuS (43) and FedX (56)
- Our source selection was faster (215ms) than HiBISCuS (400ms) and FedX (720ms)

Results source selection : overall query result



Determining more accurately the relevant sources allowed us to compute the queries' results as fast or faster than HiBISCuS and FedX

Perspectives

Life science is an **ideal domain** for developing **generic solutions**

Develop new data analysis methods

- challenges at each complexity level
- by simplifying intrinsic complexity, we probably miss some connections
  - currently : monomodal preprocessing before integration and reasoning
  - ignores the underlying biological dependencies

Address the computational challenges

Adapt data management

Life science is an **ideal domain** for developing **generic solutions**

Develop new data analysis methods

Address the computational challenges

- query performances
  - at the endpoint level
  - for federate queries
- symbolic annotations and SW provide a relevant framework, but will it be enough ?

Adapt data management

Life science is an **ideal domain** for developing **generic solutions**

Develop new data analysis methods

Address the computational challenges

Adapt data management

- adoption by end-users
- workflows
- quality and reproducibility