Protein Graph Database

Borna Bešić, Dilmurat Yusuf

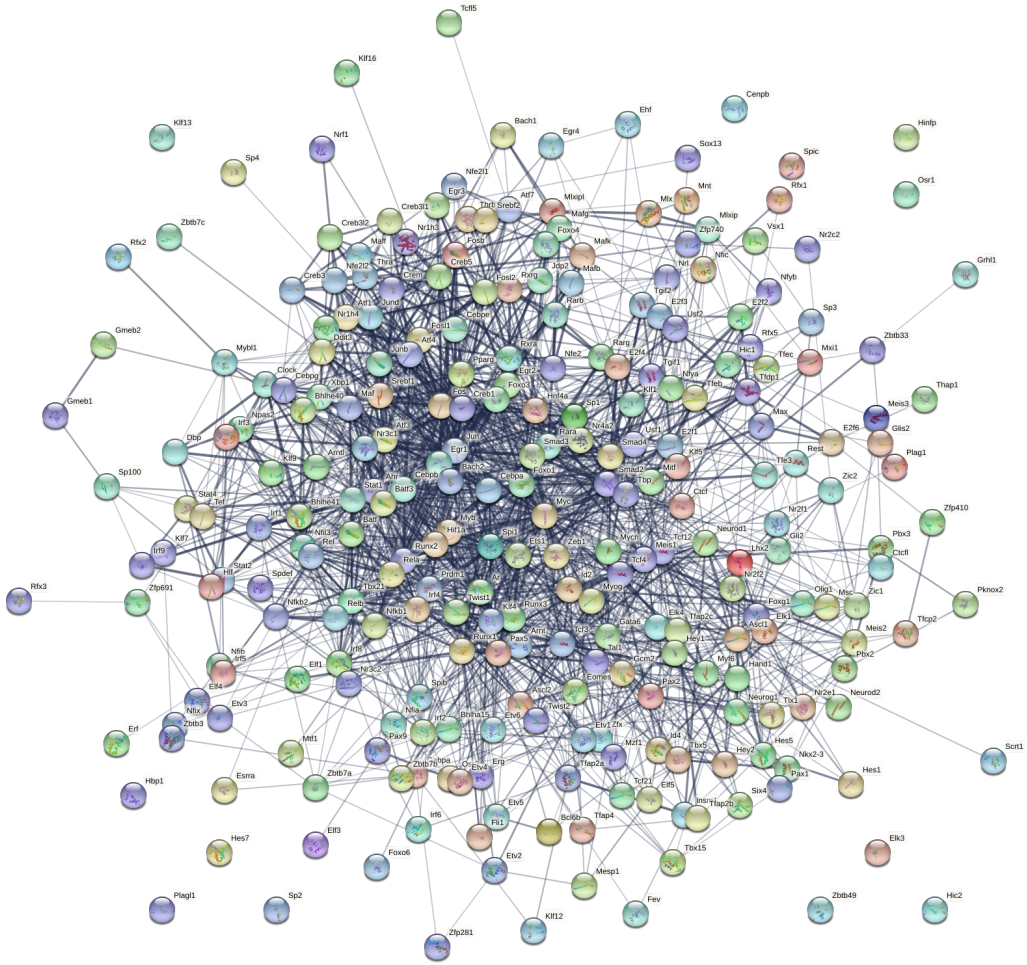
Bioinformatics Group

Albert-Ludwigs-Universität, Freiburg

May 13, 2019

Overview

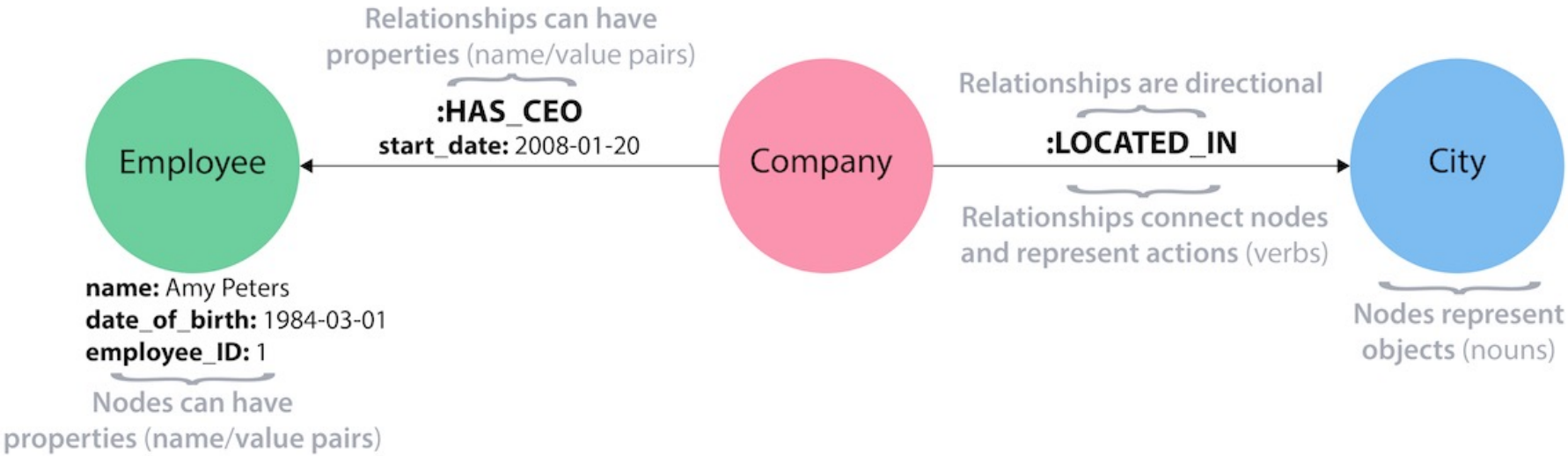
Protein network



Neo4j graph database

Cypher

```
CREATE (emp:Employee {name: "Amy Peters", date_of_birth: "1984-03-01", employee_ID: 1})
CREATE (com:Company)
CREATE (cty:City)
CREATE (com)-[:HAS_CEO {start_date: "2008-01-20"}]->(emp)
CREATE (com)-[:LOCATED_IN]->(cty)
```



Neo4j graph database

- Pros:
  - Outperforming RDBMS for associative data
  - No redundancy
  - Easier to implement a design change

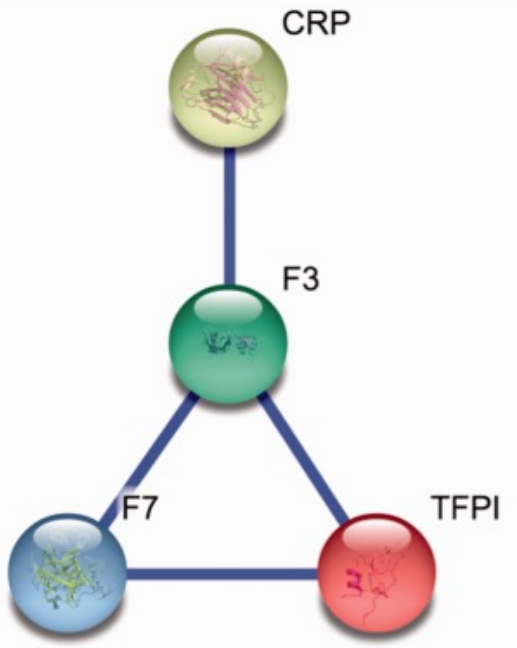
- Cons:
  - Not standard; new query language (Cypher)
  - Harder to do summing queries and max queries efficiently

Neo4j graph database

Intuitive and concise query

interactions:

p1	p2	confidence
CRP	F3	0.947
F3	F7	0.999
F3	TFPI	0.993
F7	TFPI	0.977



SQL:
```
SELECT i2.p2 FROM interactions AS i1
INNER JOIN interactions AS i2 ON i1.p2=i2.p1
WHERE i1.p1='CRP'
```

Cypher:
```
START p1=node:names(name: 'CRP')
MATCH p1-->()-->p2
RETURN p2.name
```

Figure: Have CT, Jensen LJ. Are graph databases ready for bioinformatics?

Neo4j graph database

Speed

	Neighbor network	Best-scoring path	Shortest path
PostgreSQL	206.31 s	1147.74 s	976.22 s
Neo4j	5.68 s[a]	1.17 s	0.40 s
Speedup	36×	981×	2441×

Figure: Have CT, Jensen LJ. Are graph databases ready for bioinformatics?

What do we want?

- A graph database of protein association
  - Intutive graph model
  - Queries: concise, intuitive and efficient
  - Dynamic web user interface

- Integration of different types of information
  - Protein association
  - Pathway, class
  - Disease, drug, compound

Overview

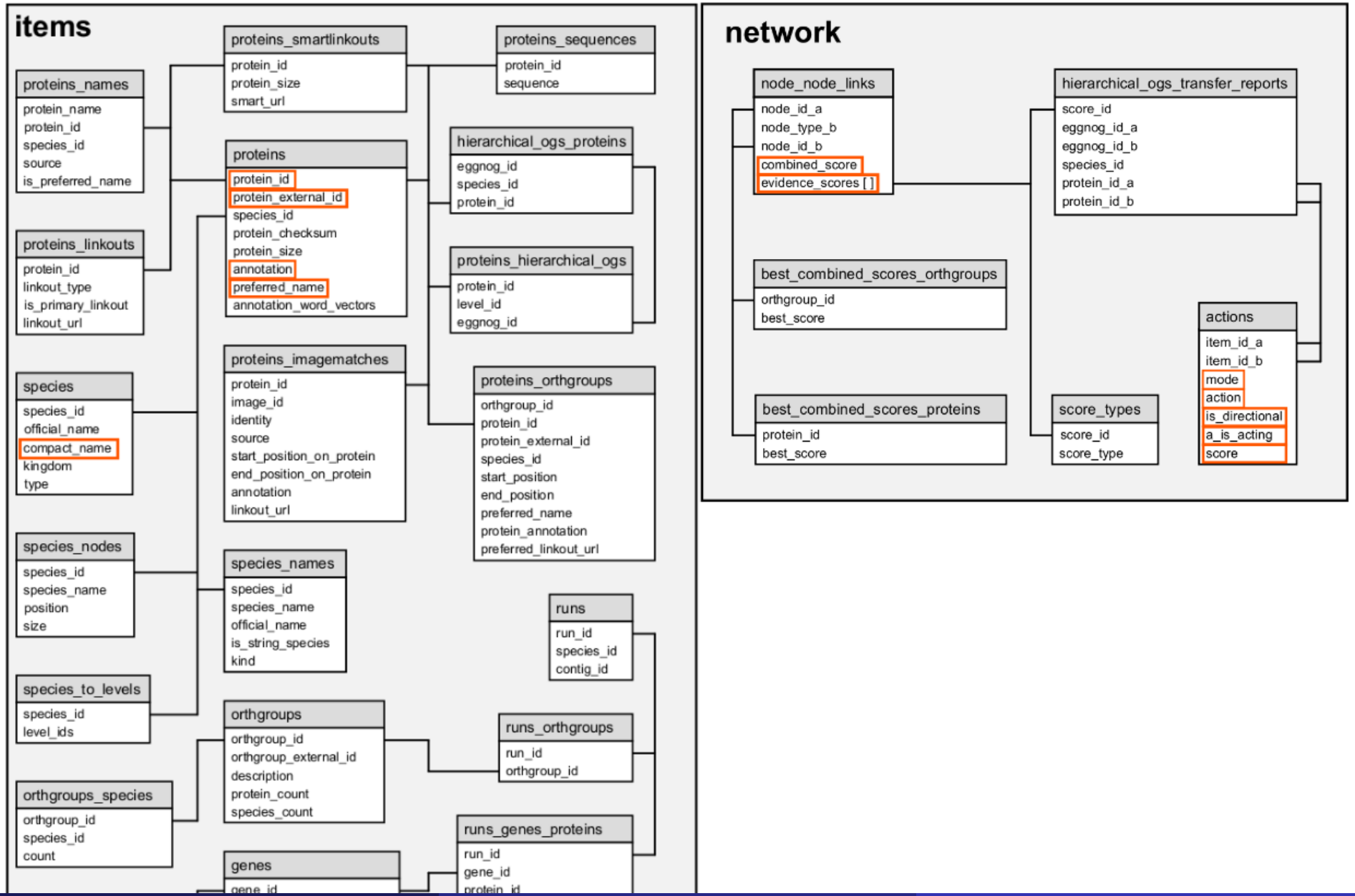
Sources of data

- STRING
  - proteins
  - protein - protein associations
- KEGG PATHWAY
  - pathways
    - classes
    - compounds
    - drugs
    - diseases

STRING

- Database of known and predicted protein-protein interactions
- 7 score channels $\rightarrow$ combined score
  1. Experiments
  2. Database
  3. Textmining
  4. Co-expression
  5. Neighbourhood
  6. Fusion
  7. Co-occurence
- One of best in overall performance for recovery of disease gene sets (Huang et al., Systematic Evaluation of Molecular Networks for Discovery of Disease Genes, Cell Systems)
- Full database dumps available for download: 512.8 GB (compressed)

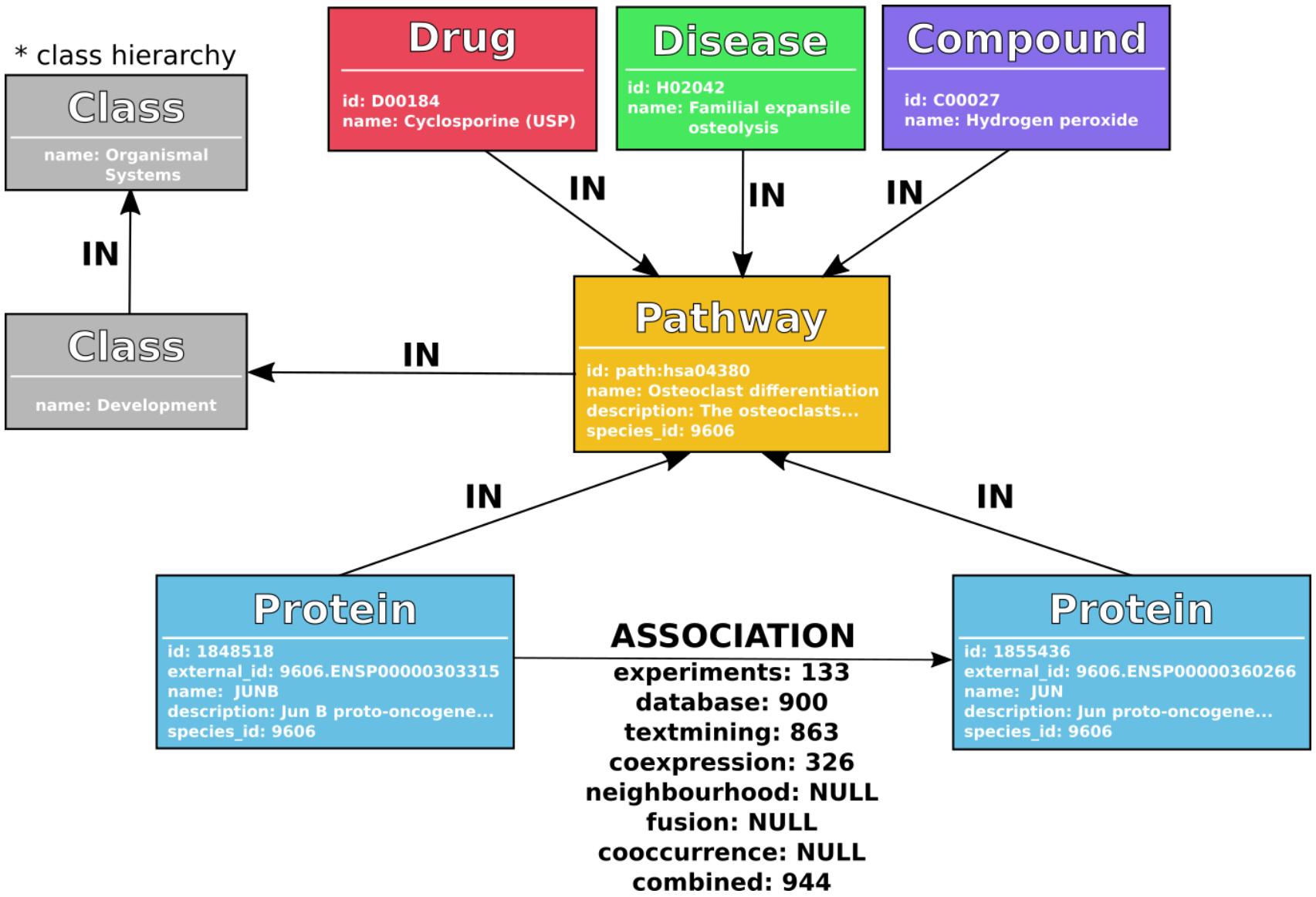
STRING Schema



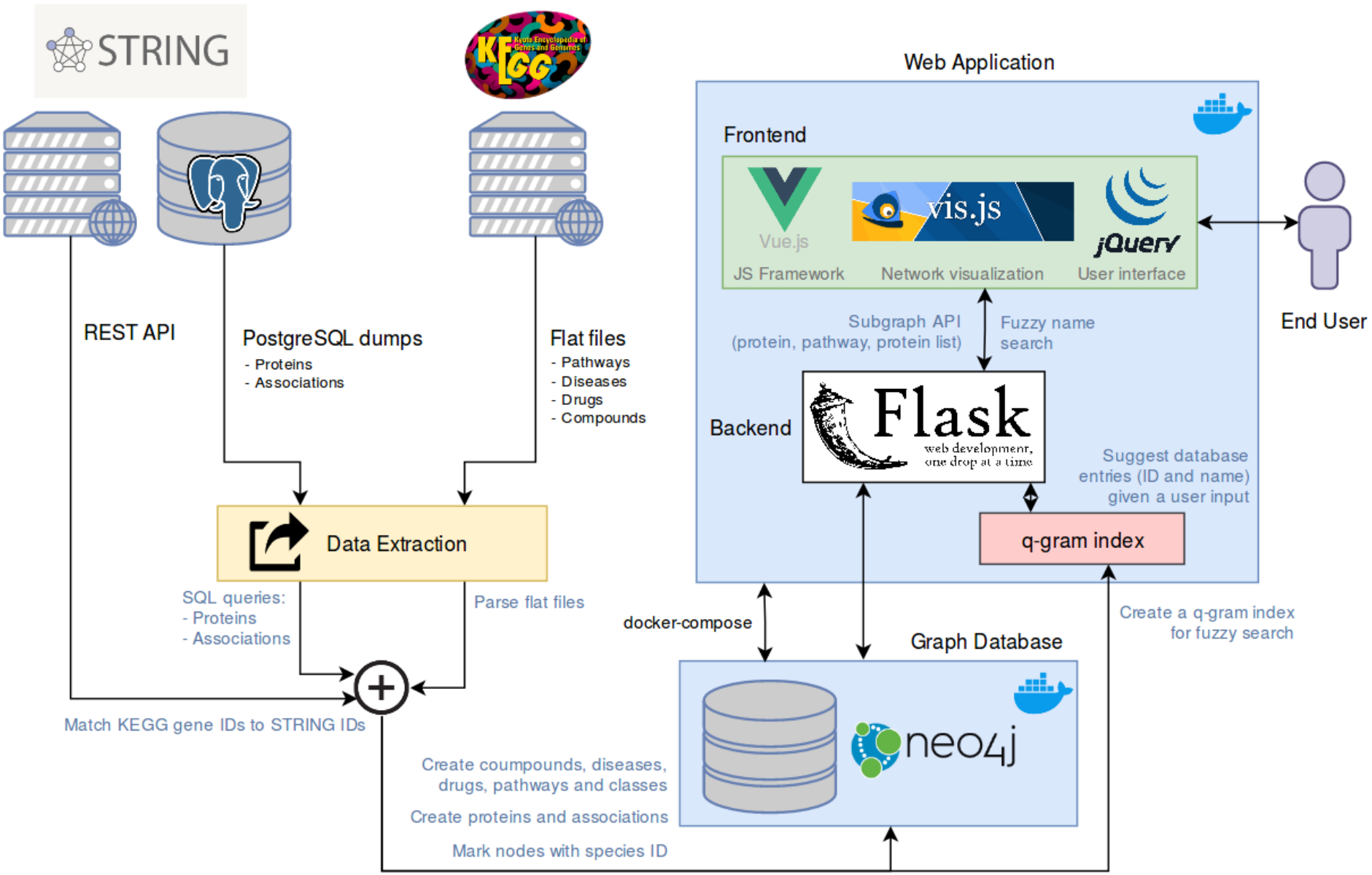
KEGG PATHWAY Flat file

```
ENTRY       hsa00010                    Pathway
NAME        Glycolysis / Gluconeogenesis - Homo sapiens (human)
DESCRIPTION Glycolysis is the process of converting glucose into pyruvate...
CLASS       Metabolism; Carbohydrate metabolism
PATHWAY_MAP hsa00010  Glycolysis / Gluconeogenesis
MODULE      hsa_M00001  Glycolysis (Embden-Meyerhof pathway), glucose => pyruvate [PATH:hsa00010]
            hsa_M00002  Glycolysis, core module involving three-carbon compounds [PATH:hsa00010]
            hsa_M00003  Gluconeogenesis, oxaloacetate => fructose-6P [PATH:hsa00010]
            hsa_M00307  Pyruvate oxidation, pyruvate => acetyl-CoA [PATH:hsa00010]
DISEASE     H00069  Glycogen storage disease
            H00071  Hereditary fructose intolerance
            H00072  Pyruvate dehydrogenase complex deficiency
            ...
DRUG        D00123  Cyanamide (JP17)
            D00131  Disulfiram (JP17/USP/INN)
            D07257  Lonidamine (INN)
            D08970  Piragliatin (USAN)
DBLINKS     BSID: 82926
            GO: 0006096 0006094
ORGANISM    Homo sapiens (human) [GN:hsa]
GENE        3101  HK3; hexokinase 3 [KO:K00844] [EC:2.7.1.1]
            3098  HK1; hexokinase 1 [KO:K00844] [EC:2.7.1.1]
            3099  HK2; hexokinase 2 [KO:K00844] [EC:2.7.1.1]
            80201  HKDC1; hexokinase domain containing 1 [KO:K00844] [EC:2.7.1.1]
            2645  GCK; glucokinase [KO:K12407] [EC:2.7.1.2]
            ...
COMPOUND    C00022  Pyruvate
            C00024  Acetyl-CoA
            C00031  D-Glucose
            C00033  Acetate
            ...
REFERENCE   (map 1)
  AUTHORS   Nishizuka Y (ed).
  TITLE     [Metabolic Maps] (In Japanese)
  JOURNAL   Tokyo Kagaku Dojin (1980)
REFERENCE   (map 1)
  AUTHORS   Nishizuka Y, Seyama Y, Ikai A, Ishimura Y, Kawaguchi A (eds).
  TITLE     [Cellular Functions and Metabolic Maps] (In Japanese)
  JOURNAL   Tokyo Kagaku Dojin (1997)
REFERENCE
  AUTHORS   Michal G.
  TITLE     Biochemical Pathways
  JOURNAL   Wiley (1999)
```

Protein Graph DB Scheme



Workflow

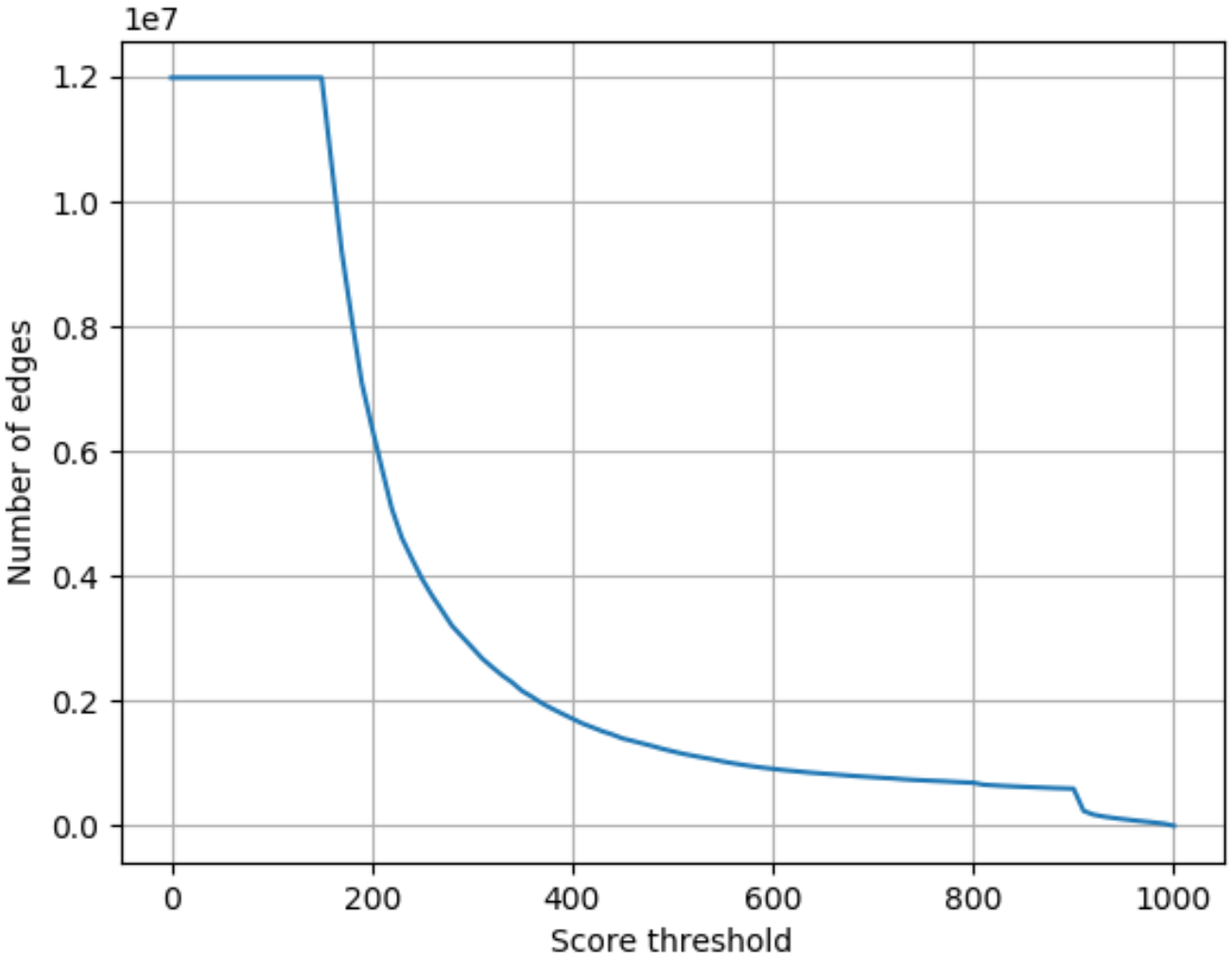

STRING
KEGG
Web Application
Frontend
Vue.js
JS Framework
vis.js
Network visualization
jQuery
User interface
End User
REST API
PostgreSQL dumps
- Proteins
- Associations
Flat files
- Pathways
- Diseases
- Drugs
- Compounds
Subgraph API
(protein, pathway, protein list)
Fuzzy name
search
Backend
Flask
web development,
one drop at a time
Suggest database
entries (ID and name)
given a user input
q-gram index
Data Extraction
SQL queries:
- Proteins
- Associations
Parse flat files
docker-compose
Graph Database
Create a q-gram index
for fuzzy search
neo4j
Match KEGG gene IDs to STRING IDs
Create coumpounds, diseases,
drugs, pathways and classes
Create proteins and associations
Mark nodes with species ID

Overview

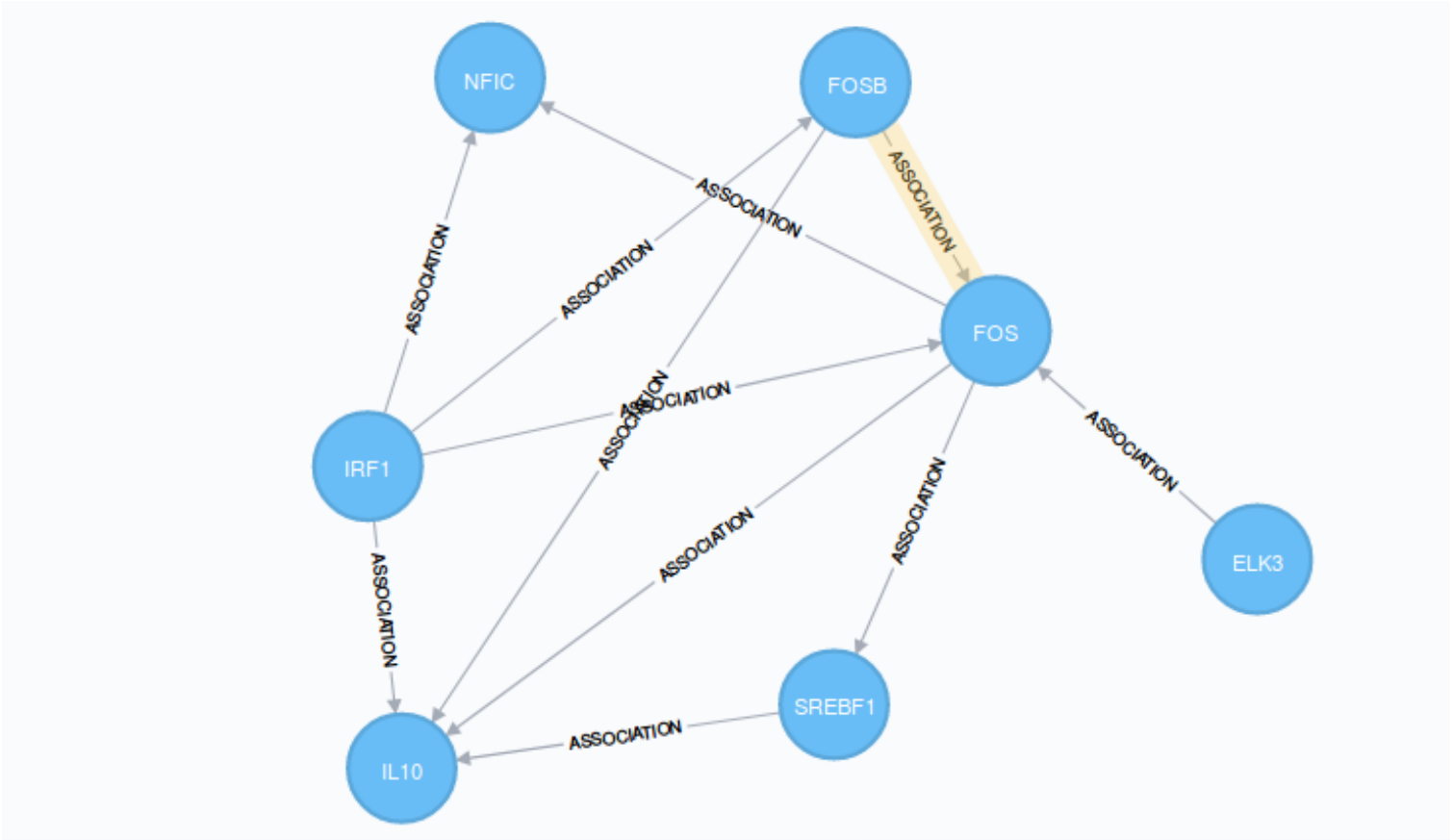
Protein Graph DB in numbers

- Species: Homo sapiens (human) & Mus musculus (house mouse)
- Nodes
  - Protein: 43 125
  - Pathway: 646
  - Drug: 3 731
  - Disease: 969
  - Compound: 3 465
  - Class: 49
- Relationships
  - **ASSOCIATION**: 11 983 549
  - **IN**: 76 715

Protein Graph DB in numbers



Protein subgraph (example)



ASSOCIATION **<id>:** 7383838 **coexpression:** 861 **combined:** 979 **database:** 800 **experiments:** 152 **textmining:** 917

Protein **<id>:** 58895 **description:** FBJ murine osteosarcoma viral oncogene homolog B; FosB interacts with Jun proteins enhancing their DNA binding activity **external_id:** 9606.ENSP00000245919 **id:** 1843783 **name:** FOSB **species_id:** 9606

Query: Central proteins

$$T_i = |\{j : j \in \mathcal{N}(i), s_{ij} \geqslant \text{threshold}\}|$$

Cypher

```
WITH ["SFPI1", "FOSB", "MLXIPL", ...] AS protein_names,
9606 AS species_id,
700 AS threshold
MATCH (p1:Protein {species_id: species_id})-[a:ASSOCIATION]-(p2:Protein)
WHERE p1.name IN protein_names AND p2.name IN protein_names AND p1.id <> p2.id
AND a.combined >= threshold
RETURN p1.name AS name, SUM(SIZE((p1)-[a]-(p2))) AS degree
ORDER BY degree DESC
```

Output

name	degree
JUN	25
FOS	19
CREB1	14
...	...

Query: Common pathways

- Common pathways given a protein list

Cypher

```
WITH ["SFPI1", "FOSB", "MLXIPL", ...] AS protein_names,
9606 AS species_id
MATCH (protein:Protein {species_id: species_id})-[element:IN]->(pathway:Pathway)
WHERE protein.name IN protein_names
RETURN DISTINCT pathway.name AS pathway, COUNT(element) AS n_proteins
ORDER BY n_proteins DESC
```

Output

pathway	n_proteins
Transcriptional misregulation in cancer	14
Human T-cell leukemia virus 1 infection	12
Pathways in cancer	12
...	...

Query: Common diseases

- Common diseases implicated by associated pathways given a protein list

Cypher

```
WITH ["SFPI1", "FOSB", "MLXIPL", ...] AS protein_names,
9606 AS species_id
MATCH (protein:Protein {species_id: species_id})-[:IN]->(pathway:Pathway)<-[:IN]-(disease:Disease)
WHERE protein.name IN protein_names
WITH disease, pathway, COUNT(protein) AS n_proteins
RETURN disease.name AS disease, COUNT(pathway) AS n_pathways, n_proteins
ORDER BY n_proteins DESC
```

Output

disease	n_pathways	n_proteins
Pituitary adenomas	1	14
Hairy-cell leukemia	1	14
Acute myeloid leukemia (AML)	1	14
...	...	...

Query: Module detection

Cypher

```
CALL algo.louvain.stream("Protein", "ASSOCIATION", {})
YIELD nodeId, community
RETURN COLLECT(nodeId) AS protein_ids, community
ORDER BY community
```

Output

protein_ids	community
[29451,29452,29453,...]	0
[29815,29899,30082,...]	1
[29858]	2
...	...

Web server (live demo)

Overview

Next steps

- Extend the database
  - Protein - protein actions
  - KEGG DRUG
  - KEGG DISEASE
  - KEGG COMPOUND
- Include other species
- Production-level web server
- Explore the possibility of a graph-native machine learning system

Thank you :)

https://backofenlab.github.io/protein-graph-database/
https://github.com/BackofenLab/protein-graph-database

- Special thanks to Prof. Rolf Backofen and Stefan Jankowski