Parco Tecnologico Padano

La ricerca si fa impresa

Filippo Biscarini

PhD, Statistical geneticist, Dipartimento di bioinformatica PTP - Lodi

A graph database to store and manage phenotypic, pedigree and genotypic data of livestock

EAAP 2013 – Nantes (France)

Caveat

Premise:

- One of the coauthors left (the main expert … argh!)

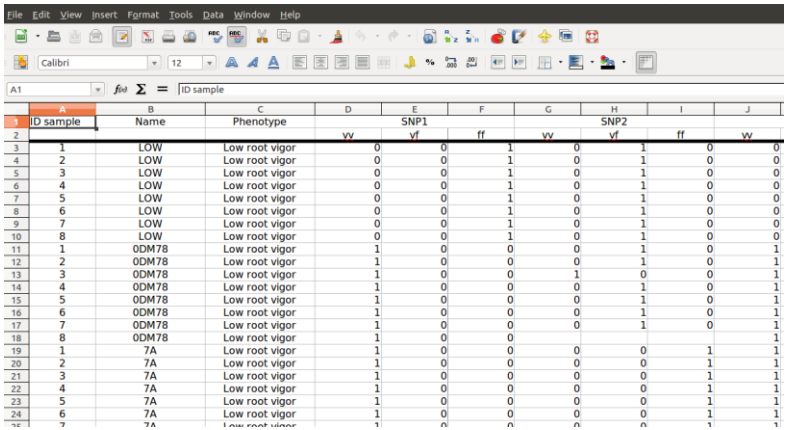
"Salvage-whatever-possible"

- ☺ tell you what this is about
- ☺ tell you how it basically works
- ☺ tell you what we have so far set up

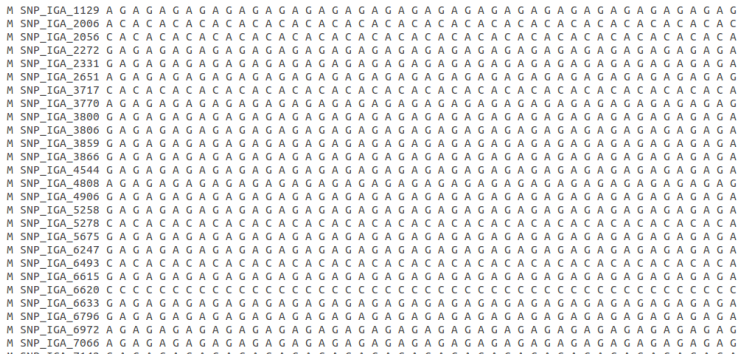
- ☹ won't be able to show any comparison in performance (it was planned but …)

- Doesn't mean this can't be taken up again (maybe someone is interested!)

Text files & spreadsheets



ID sample	Name	Phenotype	SNP1			SNP2			
			vv	vf	ff	vv	vf	ff	vv
1	LOW	Low root vigor	0	0	1	0	1	0	0
2	LOW	Low root vigor	0	0	1	0	1	0	0
3	LOW	Low root vigor	0	0	1	0	1	0	0
4	LOW	Low root vigor	0	0	1	0	1	0	0
5	LOW	Low root vigor	0	0	1	0	1	0	0
6	LOW	Low root vigor	0	0	1	0	1	0	0
7	LOW	Low root vigor	0	0	1	0	1	0	0
8	LOW	Low root vigor	0	0	1	0	1	0	0
1	0DM78	Low root vigor	1	0	0	0	1	0	1
2	0DM78	Low root vigor	1	0	0	0	1	0	1
3	0DM78	Low root vigor	1	0	0	1	0	0	1
4	0DM78	Low root vigor	1	0	0	0	1	0	1
5	0DM78	Low root vigor	1	0	0	0	1	0	1
6	0DM78	Low root vigor	1	0	0	0	1	0	1
7	0DM78	Low root vigor	1	0	0	0	1	0	1
8	0DM78	Low root vigor	1	0	0				1
1	7A	Low root vigor	1	0	0	0	0	1	1
2	7A	Low root vigor	1	0	0	0	0	1	1
3	7A	Low root vigor	1	0	0	0	0	1	1
4	7A	Low root vigor	1	0	0	0	0	1	1
5	7A	Low root vigor	1	0	0	0	0	1	1
6	7A	Low root vigor	1	0	0	0	0	1	1



country	cows	Pedigree	records
Germany	~ 4 million	~ 20 million	> 300 million

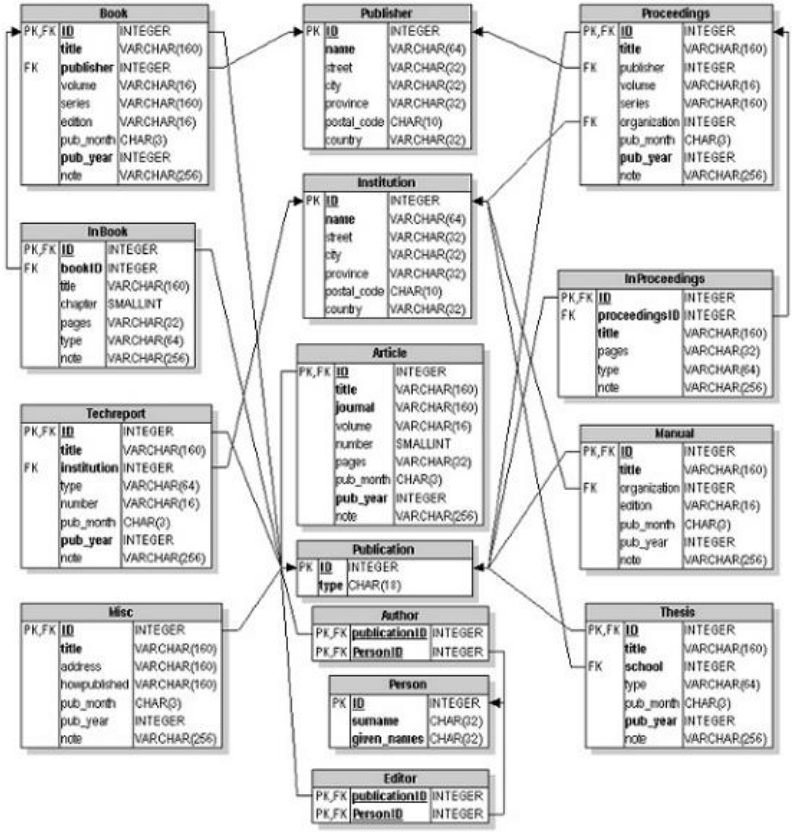
- parse each time (e.g. by chromosome, subset of animals etc …)
- less stable and less safe
- increasingly impractical with big data (e.g. HD SNP-chips, whole-genome sequences etc …)

Relational databases

- Relational databases
  - Tables
  - Relations (between tables)
- SQL (queries instead of writing a script)

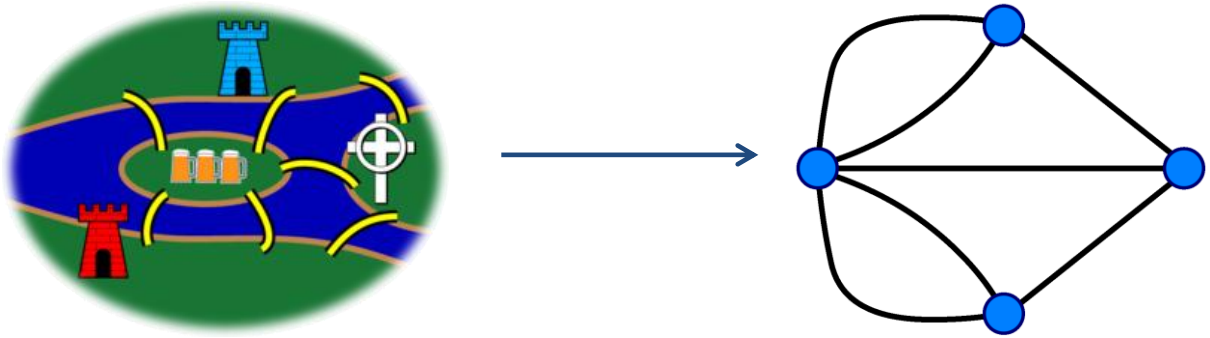
```
SELECT snp_genotype FROM
genotypes_table WHERE
chromosome=1;
```

1. more practical
2. faster
3. safer and more stable



Graph theory

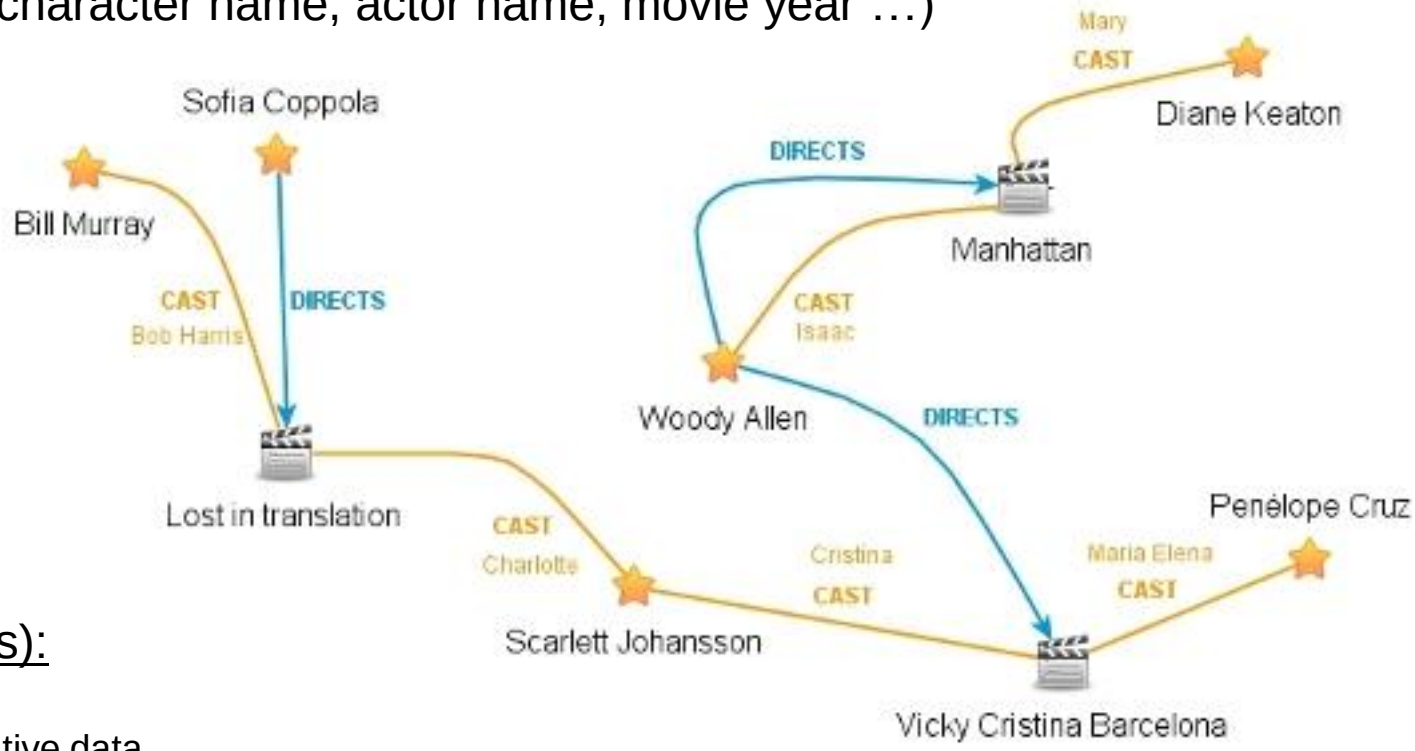
- **Euler** and the **7 bridges of Königsberg**
- *"Solutio problematis ad geometriam situs pertinentis".* Commentarii Academiae Scientiarum Imperialis Petropolitanae 8 (1736) 128- 140.



- **Graphs: nodes** and **connections**
- common models for natural and human structures
  - computer networks
  - molecules (atoms and chemical bonds)
  - habitats and migration paths (breeding patterns, spread of diseases or parasites …)

Graph database

• **nodes** (e,g, movies, directors, actors), **connections** (e.g. cast, direction) and **properties** (e.g. character name, actor name, movie year …)



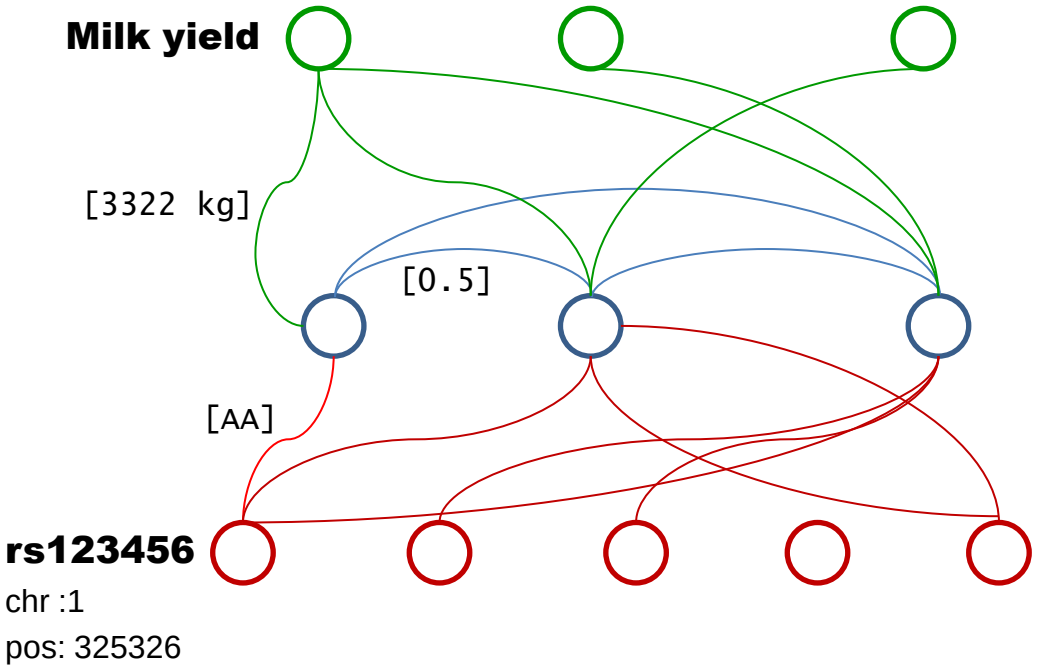
Features (promises):

- no indexes
- suited for associative data
- faster (at least for graph-like queries)
- naturally translates into object-oriented programming
- scales better to large datasets
- accommodates better changing data

Our graph database

Working example

- animal dataset (buffaloes): phenotypes, genotypes and pedigree
- 3 layers: traits, animals, markers



traits

animals (kinship value on pedigree connections)

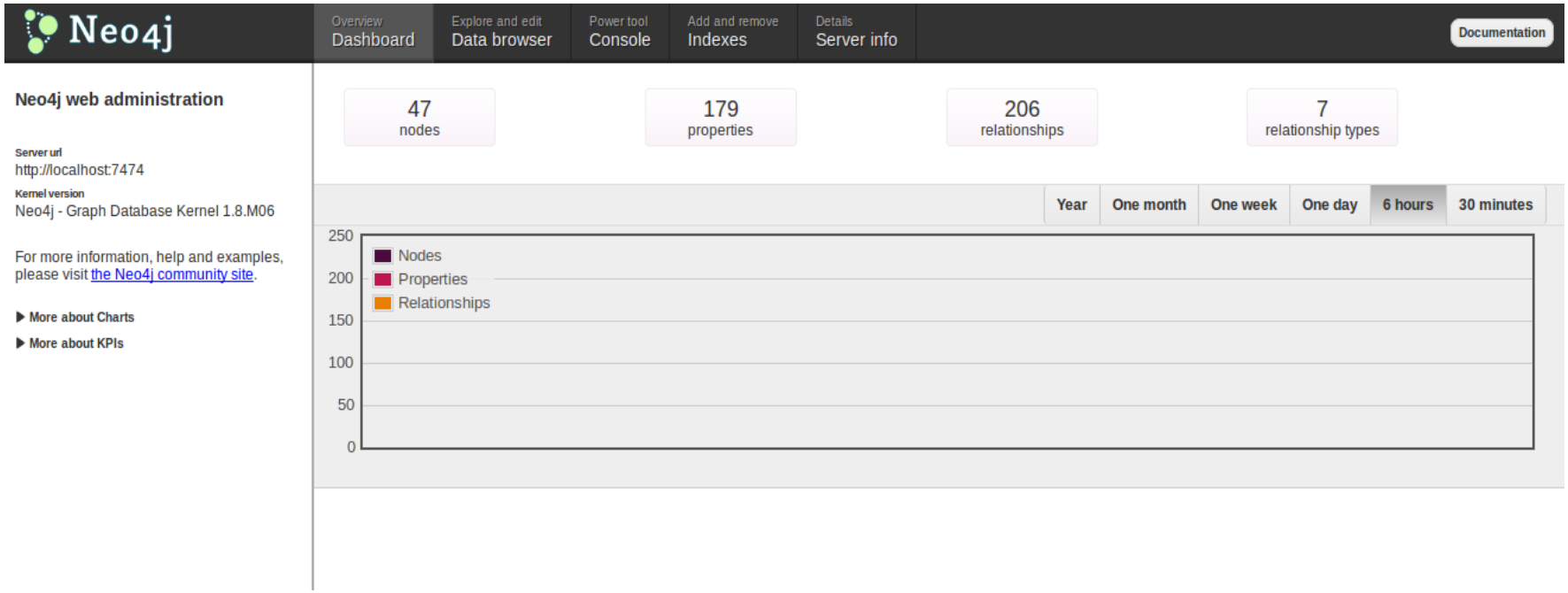
markers

Setting up the graph database

- Graph DBs are still an active field of research
- Not yet a standard query language (like SQL for relational databases)
- Lack of mature commercial products and user-friendly interfaces
- Several different ongoing projects

  - **neo4j** (http://www.neo4j.org/)
  - Ruby (create and populate the graph database)
  - Json (format for data interchange –associative array or hash)

Web Interface



Neography

Ruby gem to traverse the graph

```
n1 = Neography::Node.load(15)
q1= n1.outgoing(:contain_maker).depth(2).include_start_node
```

#<Neography::NodeTraverser:0x686baa51 @order="depth first", @filter={"language"=>"builtin", "name"=>"all"}, @relationships=[{"type"=>"contain_maker", "direction"=>"out"}], @depth=2, @uniqueness="none", @from=#<Neography::Node position="85031448", name="AX-85040742", chromosome="15">>

[parse the hash with Ruby]

Marker name: AX-85040742

Marker chromosome: 15

Marker position: 85031448

Cypher

- declarative query language to search and update the graph (no need to traverse the graph structure writing a script)
- still growing and maturing.
- some keywords (e.g.WHERE, ORDER BY) are inspired by SQL

```
START animal=node:node_auto_index(ID = 'ITM123456789')
MATCH animal[:related]->()-[:related]->related_to
RETURN animal, related_to
```

START a=node(4) RETURN a

Graph visualization

Conclusions

Breeding and genomics data are associative

Scaling is an issue (big data)

Graph databases may offer a convenient alternative to store and manage data in animal breeding and genomics

Relatively recent area of research

R&D needed before a “commercial” product can be obtained

Compare with relational DBs in terms of performance, ease of use etc …