SPARQL, a query language for RDF databases

Silvio Peroni
silvio.peroni@unibo.it – https://orcid.org/0000-0003-0530-4305 – @essepuntato

Data Science (A.Y. 2021/2022)
Second Cycle Degree in Digital Humanities and Digital Knowledge
Alma Mater Studiorum - Università di Bologna

DIPARTIMENTO DI FILOLOGIA CLASSICA E ITALIANISTICA

Summary of the previous lectures (1/2)

A **datum** is a declarative statement subject-predicate-object that, through the predicate, either **attributes** a **literal** (i.e. a value such as a string, a number, etc.) to a subject **entity** *or* it **relates** such a subject **entity** with another **entity**

Each entity, being used either as subject or object of a statement, is characterised by **a unique identifier**

The **same entity** can be used as subject or object in one or more data, while a literal **cannot be used** as subject in any datum

An attribute is intrinsically **part of** the entity to which it is associated – modifying the value of an attribute affect **only** the entity to which it refers to

A **data model** is an abstract, simplified and formal representation of some data related to a system or a real domain, and enables us to describe what a data collection is about and to check data correctness

A data model permit one to specify **classes** of entities, their **attributes** and **relations**

Summary of the previous lectures (2/2)

Depending on the structure in which data are stored (or exposed), you need to approach the queries to datasets from a different angle

- With **tabular data**, often you have to combine tables between them to obtain bigger tables which contain the query requirements and the related answer
- With **graph data**, you explore the graph starting from fixed points (i.e. known entities, values, predicates) to find a pattern that is compliant with the query

A **database** as a **collection of data** which organised, stored and accessed electronically, which can be created through a database management system (DBMS)

A **transaction** is a unit of work performed (compliant with **ACID properties**) within a DBMS against a database and usually represents any change in a database

SQL is a **query language** used and designed for managing data in a **relational** database management system, and allows one to **create** tables, to **populate** them, and finally to **query** them using a particular syntax

Any question about the previous lecture?

SPARQL

The SPARQL Protocol and RDF Query Language (SPARQL, and yes: it is a recursive name) is a query language used and designed for managing RDF data, as those available in an RDF triplestore

It is a standard that is implemented in all RDF-based triplestore, and allows one to populate and query them using a particular syntax

The full specification of the language is a W3C Recommendation and it is entirely available at https://www.w3.org/TR/sparql11-query/

A good tutorial is available at:
https://docs.data.world/tutorials/sparql/primer_intro.html

Basic SPARQL syntax for queries

```
SELECT <variables>

WHERE {
  <triple pattern 1> .
  <triple pattern 2> .
  …
}
```

A space-separated list of **?x** variables to have as result of the query (“*” means *all the variables*)

A series of conditions involving the **?x** variables above used in one or more…

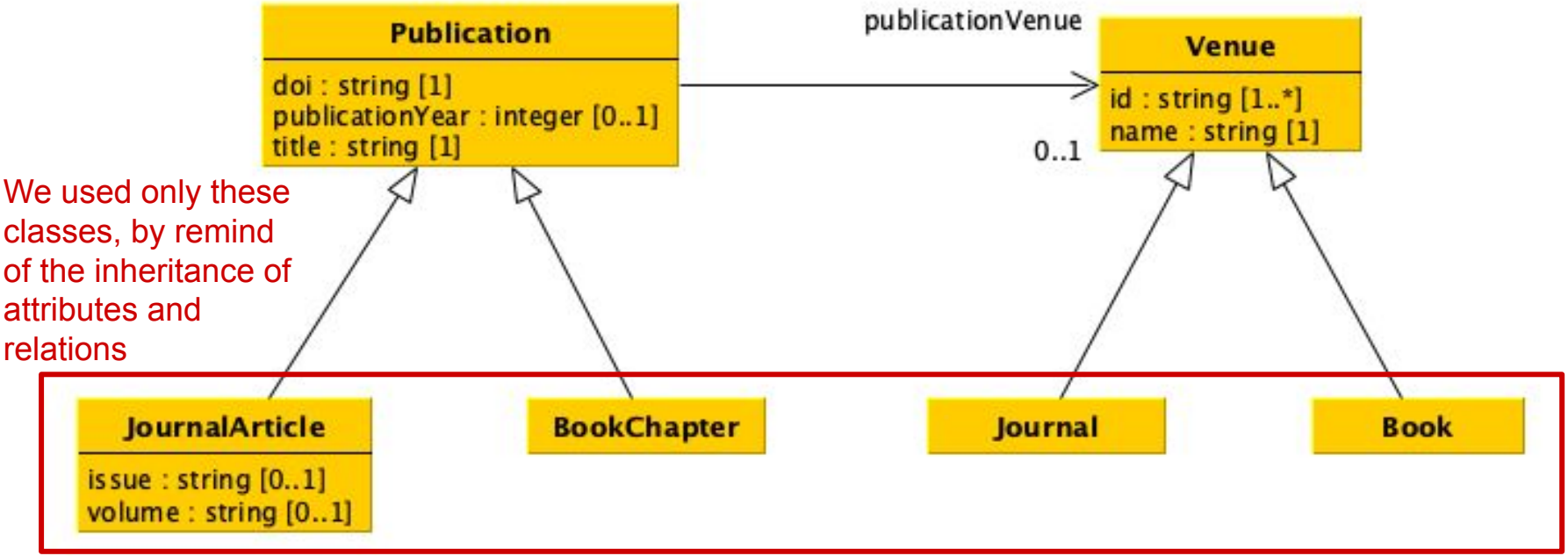
… triple patterns **S P O** where:

- **S** can be a **<url>** resource or a **?x** variable
- **P** can be a **<url>** property or a **?x** variable
- O can be a <url> resource, a **"literal"** or a **?x** variable

Questions to remind for designing a query:

- Which resources are known?
- Which literals are known?
- Which properties are known?

Exemplar classes and properties for queries

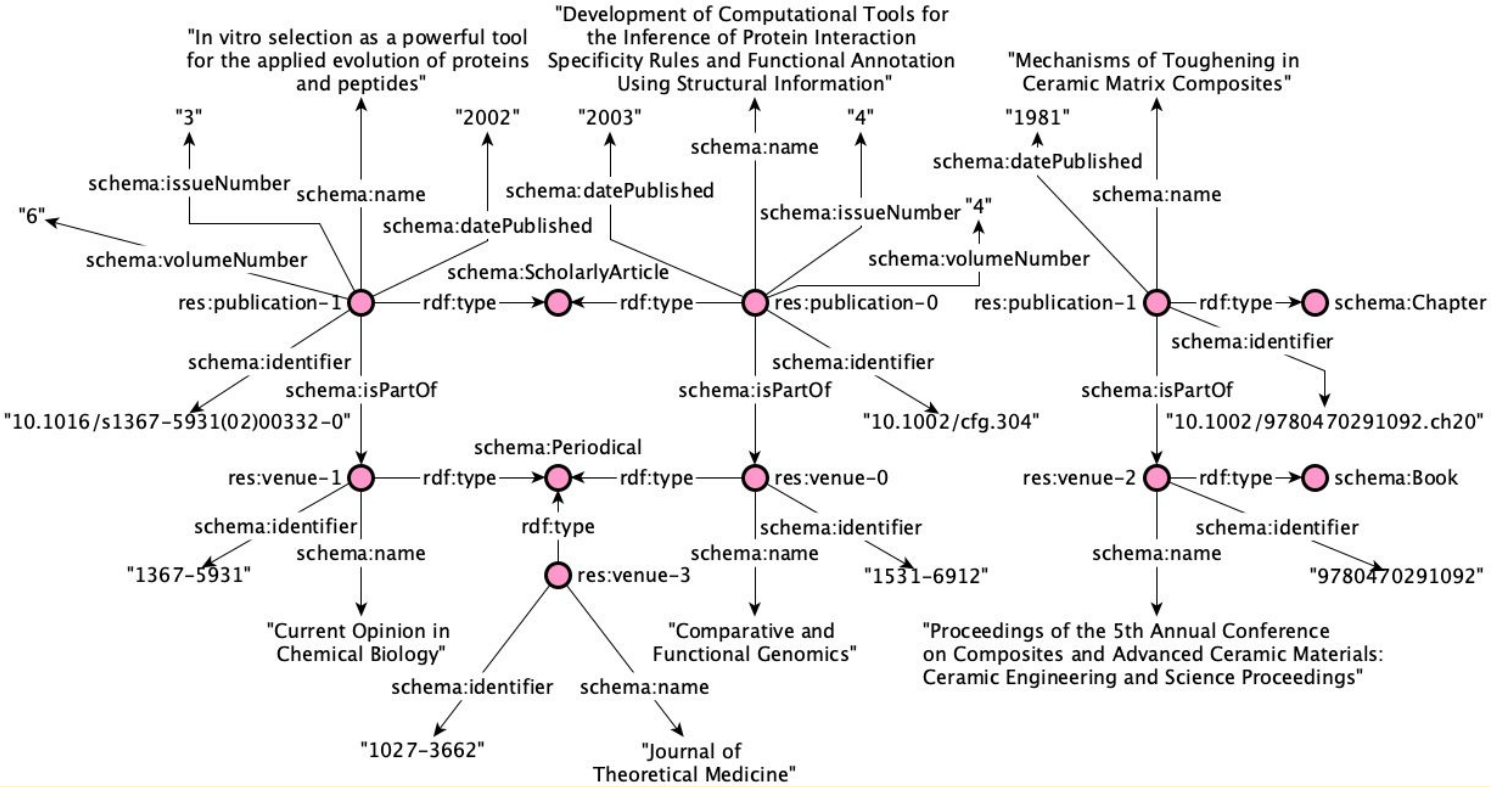


Remind that we used strings to store years of publication instead of integers

Data in the RDF graph

Prefixes

```
PREFIX res:    <https://comp-data.github.io/res/>
PREFIX rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX schema: <https://schema.org/>
```



Queries

Retrieve complete information about all journal articles

Retrieve the titles of all journal articles

Retrieve the title of the journal article with DOI “10.1016/s1367-5931(02)00332-0”

Retrieve the title of the publication with DOI “10.1016/s1367-5931(02)00332-0”

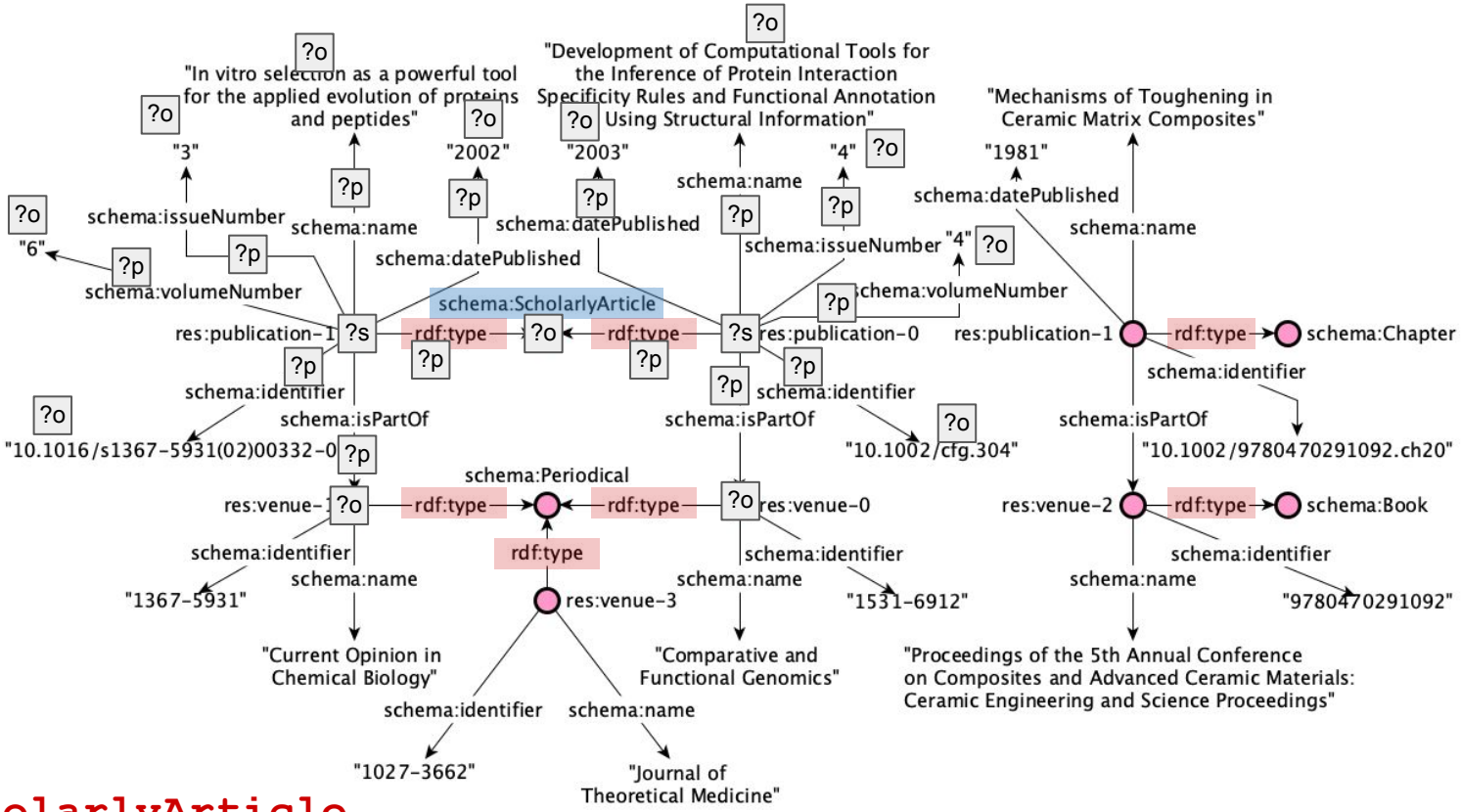
Return the name of the journal of the article with DOI “10.1016/s1367-5931(02)00332-0”

Return the id and name of the journal of the article with DOI “10.1016/s1367-5931(02)00332-0”

Query 1

Retrieve complete information about all journal articles

- Which resources are known?
- Which literals are known?
- Which properties are known?

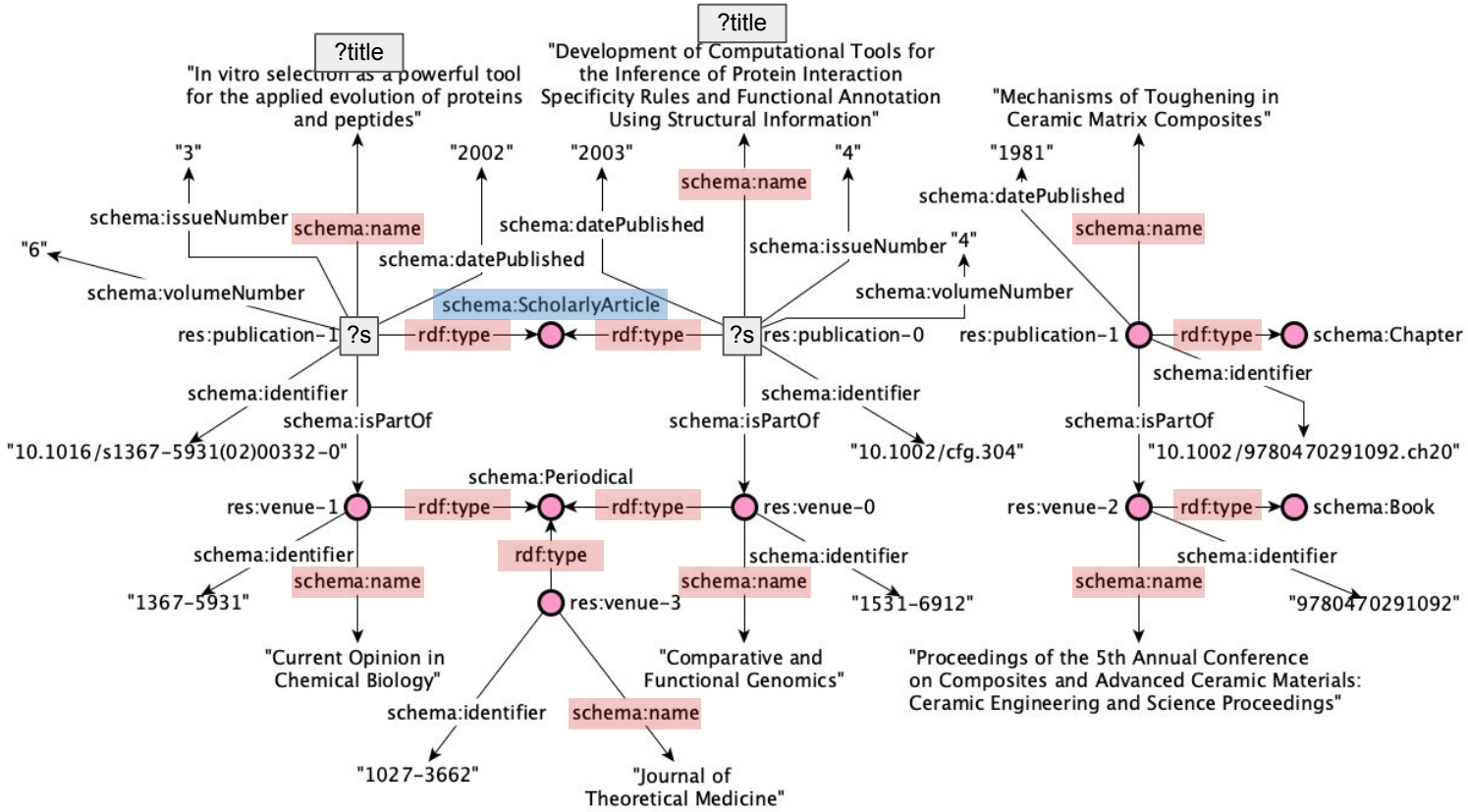



```
SELECT *
WHERE {
  ?s rdf:type
     schema:ScholarlyArticle .
  ?s ?p ?o .
}
```

Query 2

Retrieve the titles of all journal articles

- Which resources are known?
- Which literals are known?
- Which properties are known?

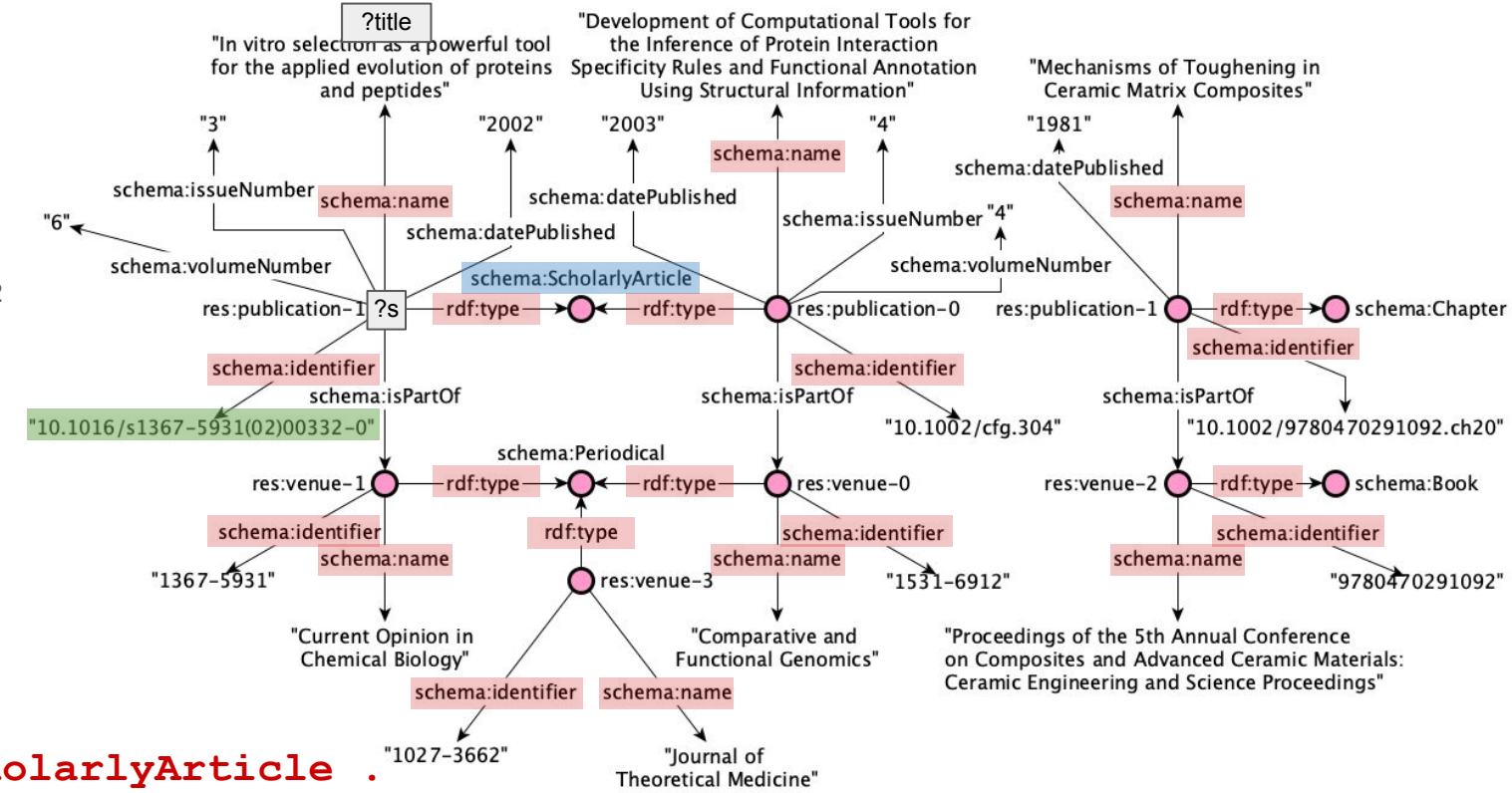



```
SELECT ?title
WHERE {
  ?s rdf:type
     schema:ScholarlyArticle .
  ?s schema:name ?title .
}
```

Query 3

Retrieve the title of the journal article with DOI “10.1016/s1367-5931(02)00332-0”

- Which resources are known?
- Which literals are known?
- Which properties are known?

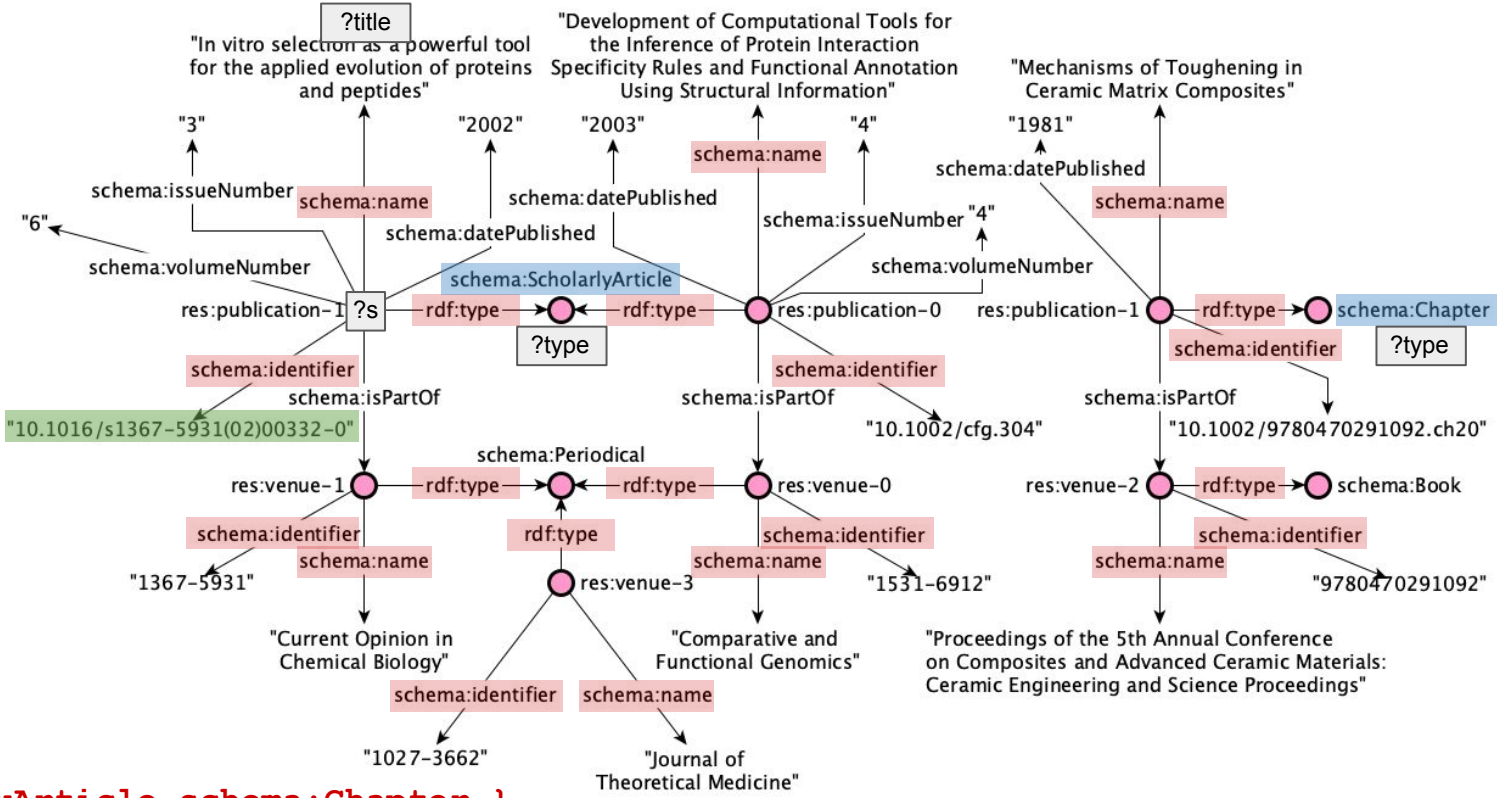



```
SELECT ?title
WHERE {
  ?s rdf:type
     schema:ScholarlyArticle .
  ?s schema:name ?title .
  ?s schema:identifier "10.1016/s1367-5931(02)00332-0" .
}
```

Query 4

Retrieve the title of the publication with DOI “10.1016/s1367-5931(02)00332-0”

- Which resources are known?
- Which literals are known?
- Which properties are known?

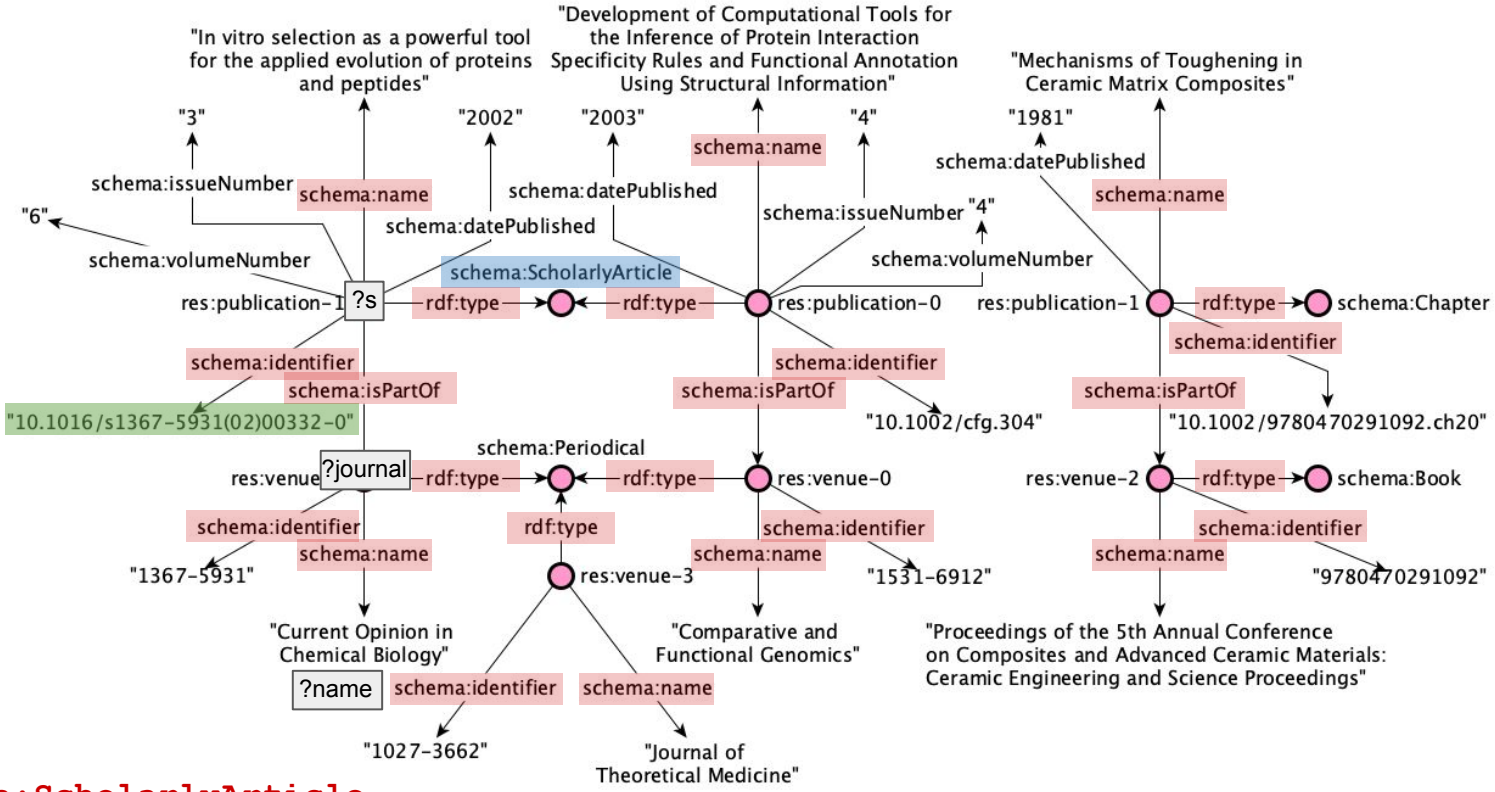



```
SELECT ?title
WHERE {
  VALUES ?type {
    schema:ScholarlyArticle schema:Chapter }
  ?s rdf:type ?type .
  ?s schema:name ?title .
  ?s schema:identifier "10.1016/s1367-5931(02)00332-0" . }
```

Query 5

Return the name of the journal of the article with DOI “10.1016/s1367-5931(02)00332-0”

- Which resources are known?
- Which literals are known?
- Which properties are known?

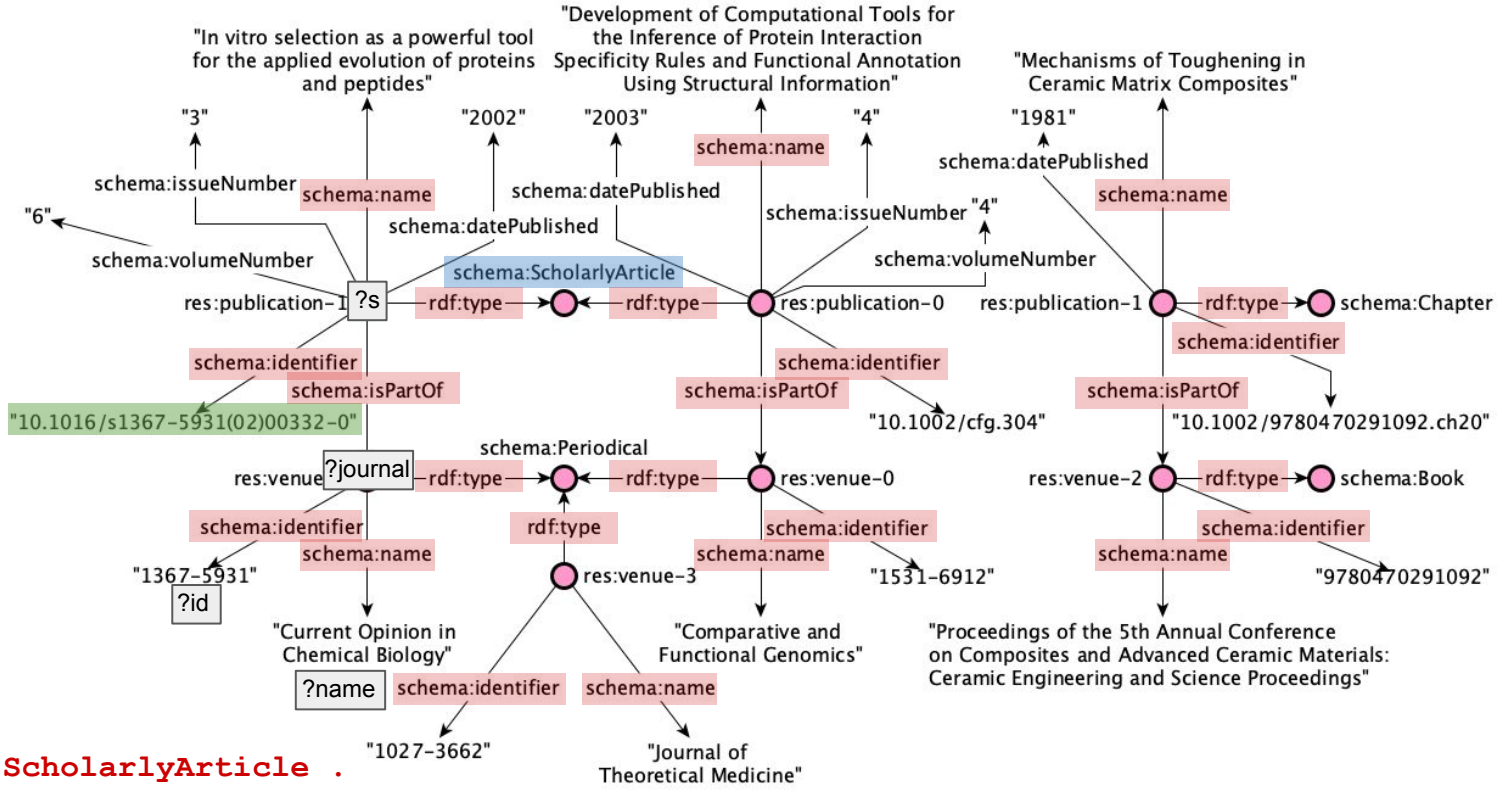



```
SELECT ?name
WHERE {
  ?s rdf:type schema:ScholarlyArticle .
  ?s schema:identifier "10.1016/s1367-5931(02)00332-0" .
  ?s schema:isPartOf ?journal .
  ?journal schema:name ?name . }
```

Query 6

Return the id and name of the journal of the article with DOI “10.1016/s1367-5931(02)00332-0”

- Which resources are known?
- Which literals are known?
- Which properties are known?




```
SELECT ?name ?id
WHERE {
  ?s rdf:type schema:ScholarlyArticle .
  ?s schema:identifier "10.1016/s1367-5931(02)00332-0" .
  ?s schema:isPartOf ?journal .
  ?journal schema:name ?name .
  ?journal schema:identifier ?id . }
```

Do you want to try them with real data?

You should have already downloaded the .jar file of Blazegraph for the previous lecture (https://github.com/comp-data/2021-2022/tree/main/docs/handson/05)

Start Blazegraph from the terminal with

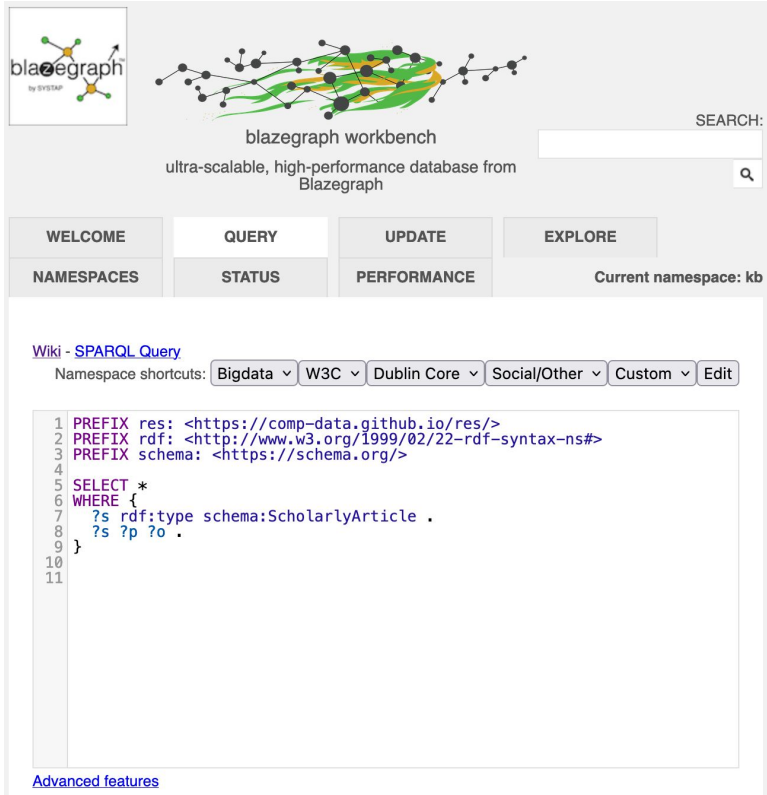
```
java -server -Xmx1g -jar blazegraph.jar
```

Open your browser and go to

```
http://127.0.0.1:9999/blazegraph/
```

Select the tab “Query” and write and execute SPARQL queries (remember the prefixes if you want to use them in the query)

Close Blazegraph from the terminal pressing CTRL+C on your keyboard



End

SPARQL, a query language for RDF databases

Silvio Peroni
silvio.peroni@unibo.it – https://orcid.org/0000-0003-0530-4305 – @essepuntato

Data Science (A.Y. 2021/2022)
Second Cycle Degree in Digital Humanities and Digital Knowledge
Alma Mater Studiorum - Università di Bologna

DIPARTIMENTO DI FILOLOGIA CLASSICA E ITALIANISTICA