

Lab 3 – Resource Description Framework (RDF) in use

1. Semantic vocabularies: Dublin Core, RDFSchema:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdfs:Class rdf:ID="MultimedialItem" />
  <rdfs:Class rdf:ID="Book">
    <rdfs:subClassOf rdf:resource="#MultimedialItem"/>
    <rdfs:label>Books class</rdfs:label>
    <rdfs:comment>Class of all books.</rdfs:comment>
  </rdfs:Class>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-obrzydlivcow">
    <dc::author rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Boris Vian</dc::author>
    <dc::title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">I wykończymy wszystkich obrzydliwców
  </dc::title>
    <dc::publisher rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Alfa</dc::publisher>
    <rdf:type rdf:resource="http://example.org/mylibrary#Book"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-rok1984">
    <dc::author rdf:datatype="http://www.w3.org/2001/XMLSchema#string">George Orwell</dc::author>
    <dc::title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Rok 1984</dc::title>
    <dc::publisher rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Muza</dc::publisher>
    <rdf:type rdf:resource="http://example.org/mylibrary#Book"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-americanpsycho">
    <dc::author rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Bret Easton Ellis</dc::author>
    <dc::title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">American Psycho</dc::title>
    <dc::publisher rdf:datatype="http://www.w3.org/2001/XMLSchema#string">vis-a-vis</dc::publisher>
    <rdf:type rdf:resource="http://example.org/mylibrary#Book"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-przenajswietrza">
    <dc::author rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Jacek Piekara</dc::author>
    <dc::title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Przenajświętrza rzeczpospolita</dc::title>
    <dc::publisher rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Fabryka Słów</dc::publisher>
    <rdf:type rdf:resource="http://example.org/mylibrary#Book"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-zly">
    <dc::author rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Leopold Tyrmand</dc::author>
    <dc::title rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Zły</dc::title>
    <dc::publisher rdf:datatype="http://www.w3.org/2001/XMLSchema#string">MG</dc::publisher>
    <rdf:type rdf:resource="http://example.org/mylibrary#Book"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://student.agh.edu.pl/~kmo/piw/mylibrary">
    <MyFavouriteBooks>
      <rdf:Bag>
        <rdf:li rdf:resource="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-rok1984"/>
        <rdf:li rdf:resource="http://student.agh.edu.pl/~kmo/piw/mylibrary#book-americanpsycho"/>
      </rdf:Bag>
    </MyFavouriteBooks>
  </rdf:Description>
</rdf:RDF>
```

2. SPARQL – demo

What semantic vocabularies are used in the queries? What are they for?

Służą do wyszukiwania specyficznych informacji w dokumentach rdf.

FOAF - opis osób,

Dublin Core – opis dokumentów,

SIOC - łączenie ze sobą serwisów dyskusyjnych,

REV - opis recenzji oraz rankingów,

PRJ – opis projektów,

RSS - powiadamianie o nowościach na stronach i wszelkiego rodzaju serwisach internetowych.

What do SELECT queries do?

Zapytanie SELECT zwraca spełniające kryteria dane zawarte w dokumencie RDF. Dane te prezentowane są w postaci tabeli, a użycie odpowiednich narzędzi, pozwala na prezentowanie tych danych w różnych formatach (JSON, CSV, XML).

What do CONSTRUCT queries do?

Zapytanie CONSTRUCT zwraca odpowiednie dane zawarte w dokumencie RDF i prezentuje je w postaci grafy RDF.

3. SPARQL queries – basics

Friends who have name and e-mail defined:

```
SELECT DISTINCT ?name, ?mail
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:mbox ?mail .
}
```

Friends who have name and e-mail defined and optional homepage:

```
SELECT DISTINCT ?name, ?mail, ?homepage
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:mbox ?mail .
  OPTIONAL { ?x foaf:homepage ?homepage }
}
```

Friends who have name and e-mail defined and optional homepage, sorted by name descending:

```

SELECT DISTINCT ?name, ?mail, ?homepage
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:mbox ?mail .
  OPTIONAL { ?x foaf:homepage ?homepage }
} ORDER BY DESC (?name)

```

4. SPARQL queries – options

People whose name starts with 'K'

```

SELECT DISTINCT ?name
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name .
  FILTER regex(?name, „^K.*”)
}

```

People who are older than 18 years old

```

SELECT DISTINCT ?name, ?age
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:age ?age .
  FILTER( ?age > 18 )
}

```

People whose name starts with 'K' or are older than 18 years old, make search case insensitive

```

SELECT DISTINCT ?name, ?age
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name.
  ?x foaf:age ?age .
  FILTER (?age > 18) || regex(?name, „^K.*”, „i”)
}

```

People having e-mails on student.agh.edu.pl server

```

SELECT DISTINCT ?name, ?mail
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:mbox ?mail
  FILTER regex(?mail, „.*student.agh.edu.pl$”)
}

```

```
}
```

Name of people, who have homepage or e-mail on student.agh.edu.pl server

```
SELECT DISTINCT ?name, ?mail, ?page
WHERE {
  ?x rdf:type foaf:Person ;
  ?x foaf:name ?name ;
  ?x foaf:mbox ?mail ;
  ?x foaf:homepage ?page ;
  FILTER regex(?mail, „.*student.agh.edu.pl$”) || regex(?page, „^http://student.agh.edu.pl.*”)
}
```

5. Open Data Sets

What are the main limitations of using (querying for information) the RDFdatasets such as DBPedia or MusicBrainz?

Główną wadą zapytań do tego typu baz wiedzy jest spory czas jego wykonania. Szczególnie w przypadku bardziej złożonych zapytań czas ten może być bardzo długi. Kolejną wadą jest ograniczenie zapytań do języka SPARQL, co znacznie obniża użyteczność narzędzia. Wymienione bazy wiedzy na dzień dzisiejszy nie posiadają jeszcze wystarczająco dużo danych, aby móc spełnić odpowiednio swoją rolę.