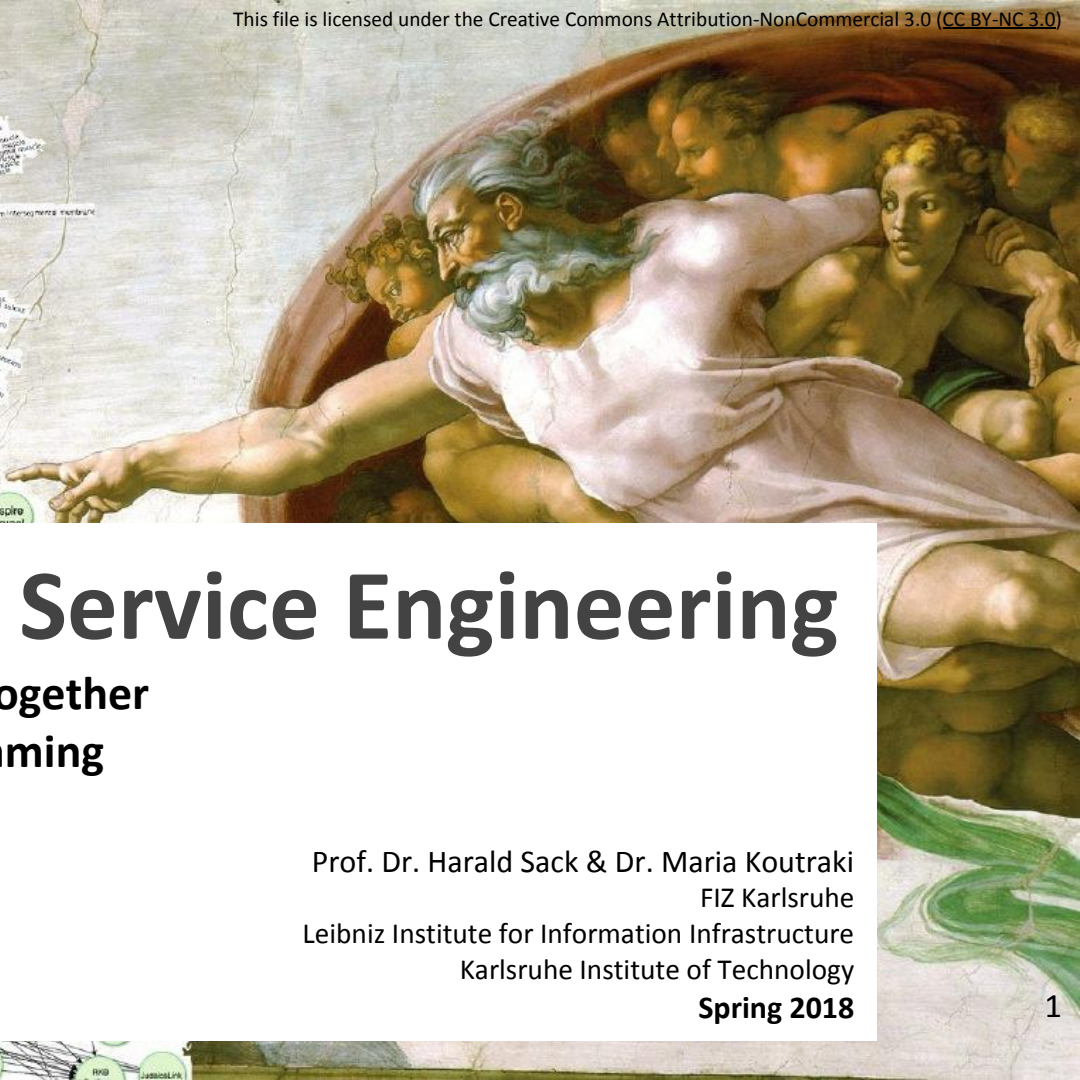




6.1 Linked Data Programming



Leibniz Institute for Information Infrastructure



Hasso
Plattner
Institut



Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Spring 2018

Information Service Engineering

Lecture 6: Putting it All Together

6.1 Linked Data Programming

6.2 Knowledge Mining and Information Extraction

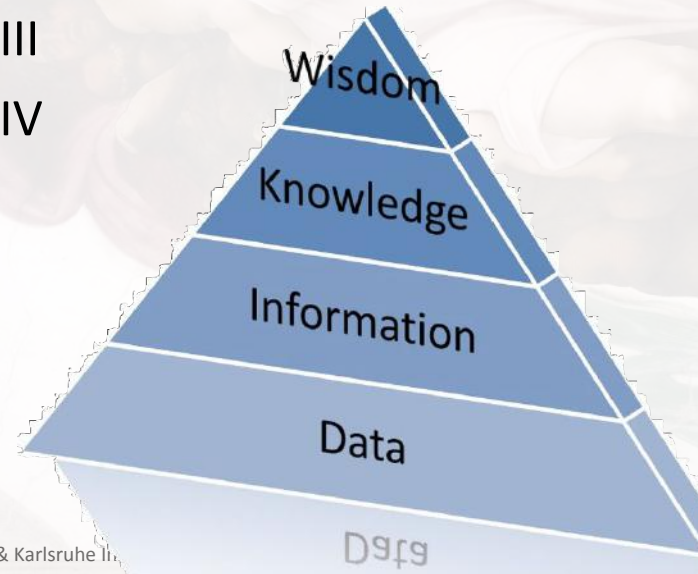
6.3 Knowledge Mining and Information Extraction II

6.4 Knowledge Mining and Information Extraction III

6.5 Knowledge Mining and Information Extraction IV

6.6 Semantic Search

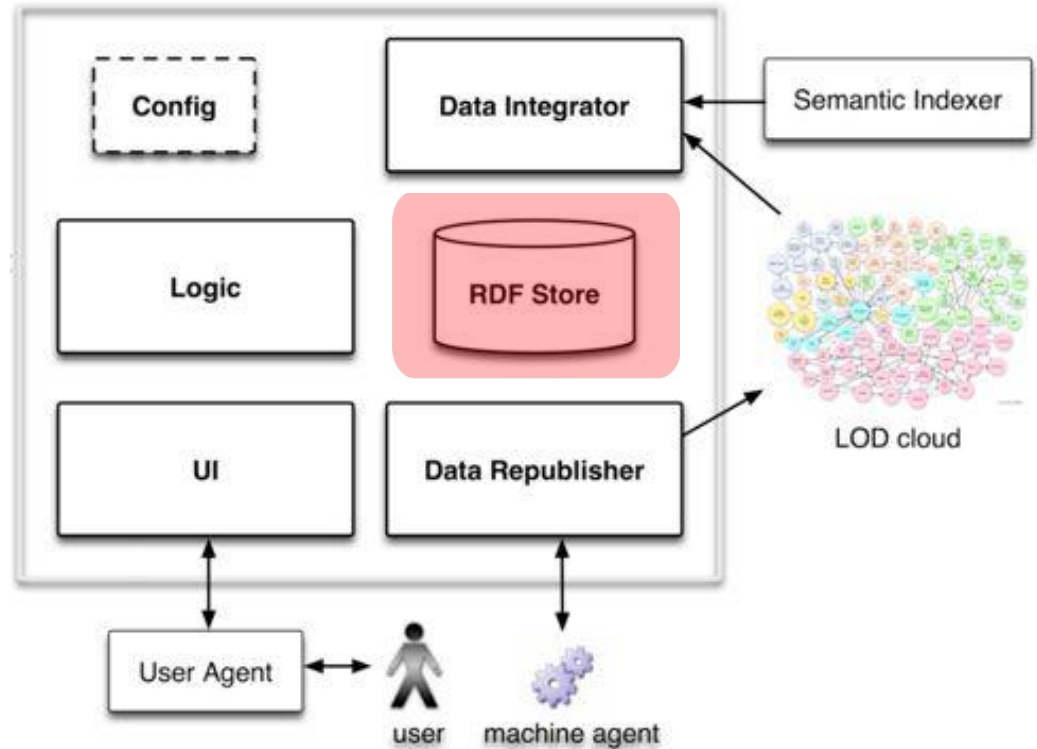
6.7 Exploratory Search



Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

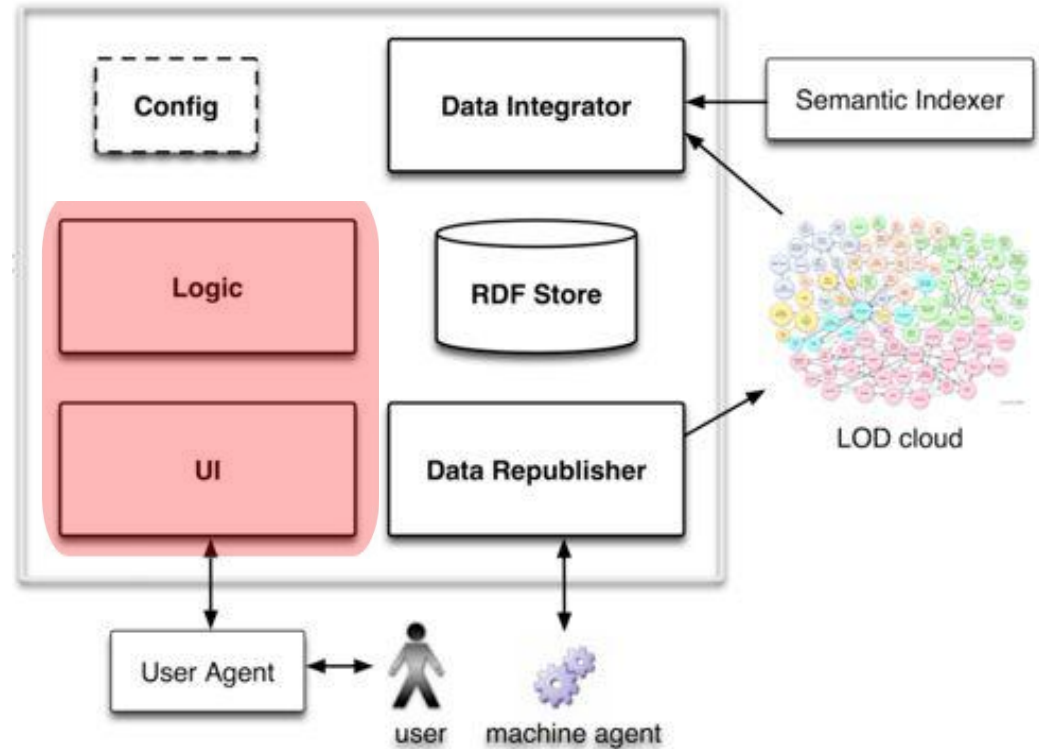
- Required Components:
 - **Local RDF Store**
 - caching of results
 - permanent storage



Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

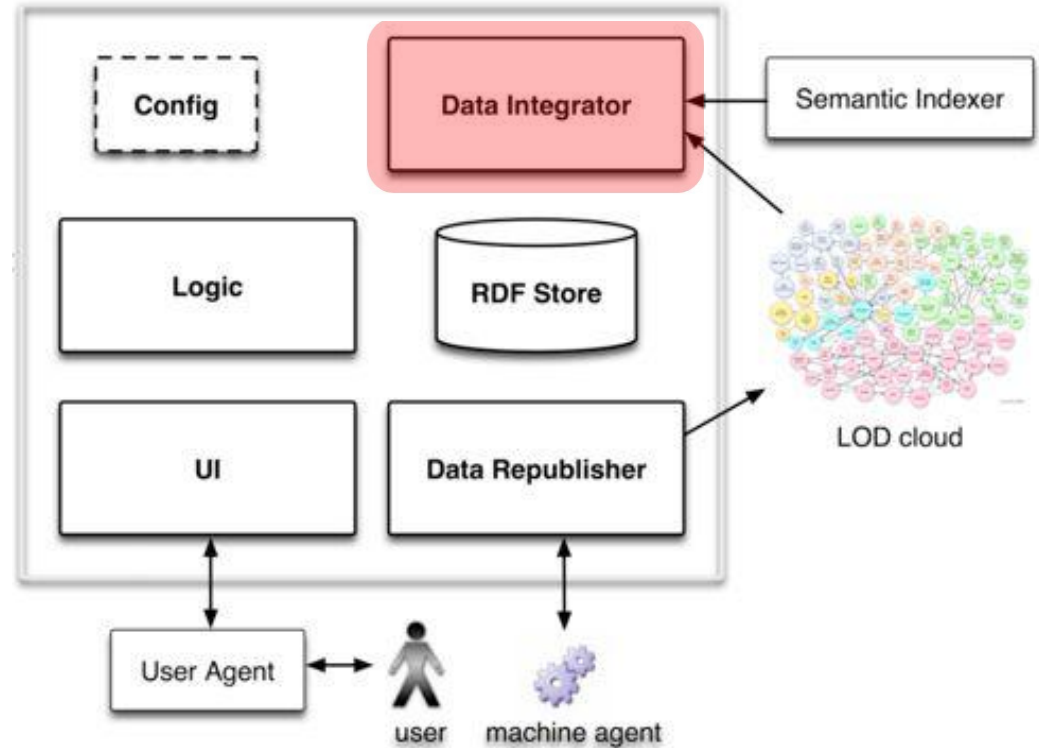
- Required Components:
 - **Logic** (Controller) and
 - **User Interface**
 - (=Business Logic)
 - (not LOD specific)



Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

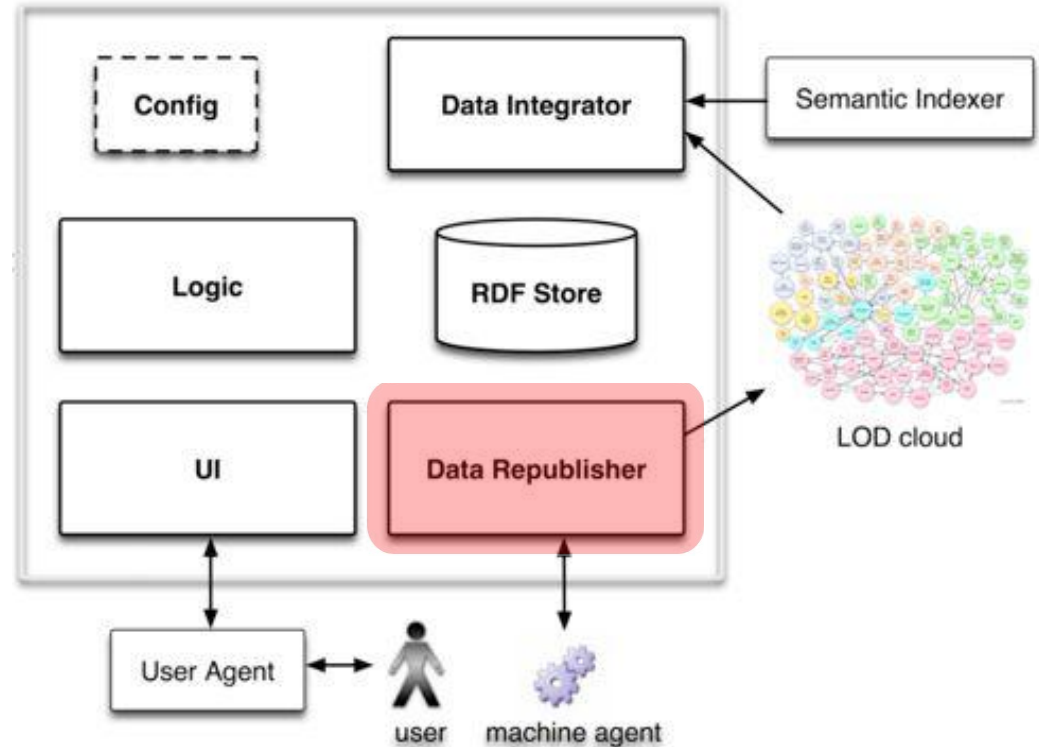
- Required Components:
 - **Data Integration component**
 - get data directly from LOD-Cloud or
 - via Semantic Indexer



Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- Required Components:
 - **Data Re-/Publishing component**
 - write back application dependent data into the Web of Data

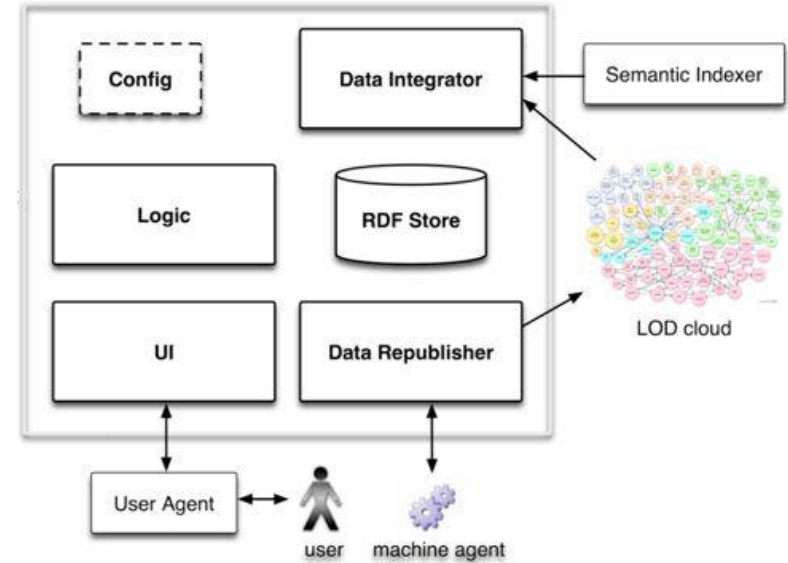


Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- Required Components:

- **Local RDF Store**
 - caching of results
 - permanent storage
- **Logic** (Controller) and
- **User Interface** (=Business Logic)
 - (not LOD specific)
- **Data Integration component**
 - get data directly from LOD-Cloud or
 - via Semantic Indexer
- **Data Re-/Publishing component**
 - write back application dependent data into the Web of Data



Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- The easiest way is to make use of a suitable library:
 - SPARQL Javascript Library
http://www.thefigtrees.net/lee/blog/2006/04/sparql_calendar_demo_a_sparql.html
 - ARC for SPARQL (PHP)
<https://github.com/semsol/arc2/wiki>
 - dotNetRDF (C#)
<https://dotnetrdf.github.io/>
 - Jena/ARQ (Java)
<http://jena.apache.org/>
 - Sesame (Java)
<http://rdf4j.org/>
 - **SPARQL Wrapper (Python)**
<http://rdflib.github.io/sparqlwrapper/>

Linked Data Driven Web Applications

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- The easiest way is to make use of a suitable library:
 - **SPARQL Wrapper (Python)**
<http://rdflib.github.io/sparqlwrapper/>
- Access to Linked Data via **SPARQL endpoints**
 - let's choose **DBpedia** (just for simplicity...)
<http://dbpedia.org/sparql>
- ...now we have to think of a simple example...

Linked Data Programming Example

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- **Example Application:**

- Build a simple application that looks for **today's birthdays** of famous people, as e.g. **scientists**
- Create a **list of scientists**, whose birthday is today including some **additional information**, as e.g.
 - Year of Birth
 - Short description, picture (if available), etc.
- Let's create a simple (local) **web page** for the task (i.e. encode results in HTML), which can be displayed in the browser
- we use **Python** and the **SPARQL Wrapper library**

Prerequisites - (Manual) Data Analysis

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- Choose a representative example:
 - E.g. **Albert Einstein** from **DBpedia**

 Browse using  Formats  Faceted Browser  Sparql Endpoint 

About: Albert Einstein

An Entity of Type : scientist, from Named Graph : <http://dbpedia.org>, within Data Space : dbpedia.org

Albert Einstein (/ˈaɪnstʌɪn/; German: [ˈalbɛʁt ˈaɪnʃtaɪn] ; 14 March 1879 – 18 April 1955) was a German-born theoretical physicist. He developed the general theory of relativity, one of the two pillars of modern physics (alongside quantum mechanics). Einstein's work is also known for its influence on the philosophy of science. Einstein is best known in popular culture for his mass–energy equivalence formula $E = mc^2$ (which has been dubbed "the world's most famous equation"). He received the 1921 Nobel Prize in Physics for his "services to theoretical physics", in particular his discovery of the law of the photoelectric effect, a pivotal step in the evolution of quantum theory.

Property	Value
dbo:abstract	Albert Einstein (/ˈaɪnstʌɪn/; German: [ˈalbɛʁt ˈaɪnʃtaɪn] ; 14 March 1879 – 18 April 1955) was a German-born theoretical physicist. He developed the general theory of relativity, one of the two pillars of modern physics (alongside quantum mechanics). Einstein's work is also known for its influence on the philosophy of science. Einstein is best known in popular culture for his mass–energy equivalence formula $E = mc^2$ (which has been dubbed "the world's most famous equation"). He received the 1921 Nobel Prize in Physics for his "services to theoretical physics", in particular his discovery of the law of the photoelectric effect, a pivotal step in the evolution of quantum theory. Near the beginning of his career, Einstein thought that Newtonian mechanics was no longer enough to reconcile the laws of classical mechanics with the laws of the electromagnetic field. This led to the development of his special theory of relativity. He realized, however, that the principle of relativity could also be extended to gravitational fields, and with his subsequent theory of gravitation in 1916, he published a paper on general relativity. He continued to deal with problems of statistical mechanics and quantum theory, which led to his explanations of particle theory and the motion of molecules. He also investigated the thermal properties of light which laid the foundation of the photon theory of light. In 1917, Einstein applied the general theory of relativity to model the large-scale structure of the universe. He was visiting the United States when Adolf Hitler came to power in 1933 and, being Jewish, did not go back to Germany, where he had been a professor at the Berlin Academy of Sciences. He settled in the U.S., becoming an American citizen in 1940. On the eve of World War II, he endorsed a letter to President Franklin D. Roosevelt alerting him to the potential development of "extremely powerful bombs of a new type" and recommending that the U.S. begin similar research. This eventually led to what would become the Manhattan Project. Einstein supported defending the Allied forces, but largely denounced the idea of using the newly discovered nuclear fission as a weapon. Later, with the British philosopher Bertrand Russell, Einstein signed the Russell–Einstein Manifesto, which highlighted the danger of nuclear weapons. Einstein was affiliated with the Institute for Advanced Study in Princeton, New Jersey, until his death in 1955. Einstein published more than 300 scientific papers and four books on physics.

http://dbpedia.org/page/Albert_Einstein

Prerequisites - (Manual) Data Analysis

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- Data Analysis via **SPARQL**

- What kind of entities are you looking for?

- `?scientist rdf:type dbo:Scientist .`

- What information do you need?

- `?scientist dbo:birthDate ?birthdate .`

- `?scientist rdfs:label ?name .`

- `?scientist rdfs:comment ?description .`

- `OPTIONAL { ?scientist dbo:thumbnail ?thumbnail }`

- Any filter criteria?

- `(lang(?name)="en") && (lang(?description)="en")`

- `(SUBSTR(STR(?birthdate),6)="07-05")`

- More sophisticated: `(SUBSTR(STR(bif:curdate('')),6)`

*Virtuoso triple
store builtin
function for
current date*

Prerequisites - SPARQL Queries

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

[SPARQL query at dbpedia.org](http://dbpedia.org)

Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)

Query Text

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?birthdate ?thumbnail ?scientist ?name ?description WHERE {
  ?scientist rdf:type dbo:Scientist ;
             dbo:birthdate ?birthdate ;
             rdfs:label ?name ;
             rdfs:comment ?description
  FILTER ((lang(?name)="en")&&(lang(?description)="en")&&
    (STRLEN(STR(?birthdate))>6)&&(SUBSTR(STR(?birthdate),6)=SUBSTR(STR(bif:curdate('')),6))) .
  OPTIONAL { ?scientist dbo:thumbnail ?thumbnail . }
} ORDER BY ?birthdate
```

(Security restrictions of this server do not allow you to retrieve remote RDF data, see [details](#).)

Results Format:

Execution timeout:

milliseconds (values less than 1000 are ignored)

Options:

- ☒ Strict checking of void variables
- ☐ Log debug info at the end of output (has no effect on some queries and output formats)
- ☐ Generate SPARQL compilation report (instead of executing the query)

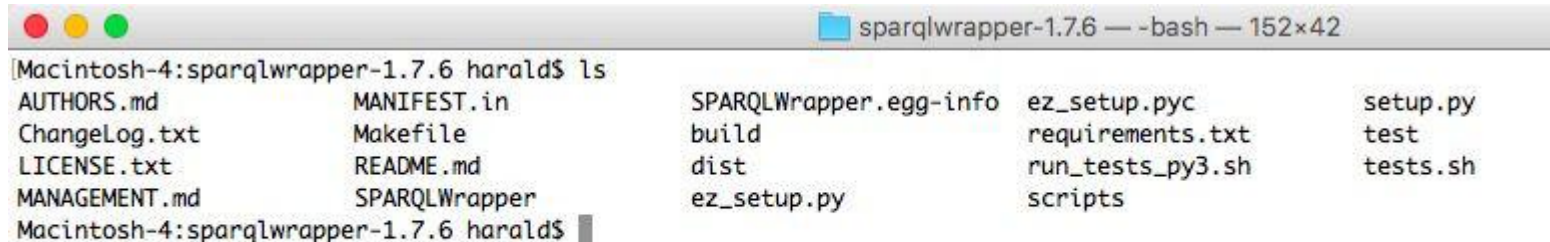
(The result can only be sent back to browser, not saved on the server, see [details](#))

Copyright © 2018 [OpenLink Software](#)
Virtuoso version 07.20.3226 on Linux (i686-generic-linux-glibc212-64), Single Server Edition

Prerequisites - Download and Install

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

- You should have Python installed on your computer
 - <https://www.python.org/downloads/>
- Download SPARQL Wrapper for Python
 - <http://rdflib.github.io/sparqlwrapper/>
- Follow the instructions for Installation



```
Macintosh-4:sparqlwrapper-1.7.6 harald$ ls
AUTHORS.md      MANIFEST.in      SPARQLWrapper.egg-info  ez_setup.pyc      setup.py
ChangeLog.txt   Makefile          build                   requirements.txt   test
LICENSE.txt     README.md         dist                    run_tests_py3.sh  tests.sh
MANAGEMENT.md   SPARQLWrapper    ez_setup.py            scripts
```

example_02.py

```
1 #!/usr/bin/python
2 # -*- coding: utf-8 -*-
3
4 import urllib2
5 from datetime import datetime
6 from SPARQLWrapper import SPARQLWrapper, JSON, XML, N3, RDF
7
8 sparql = SPARQLWrapper("http://dbpedia.org/sparql")
9
10 sparql.setQuery("""
11 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
12 PREFIX dbo: <http://dbpedia.org/ontology/>
13 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
14 PREFIX dc: <http://purl.org/dc/elements/1.1/>
15
16 Select distinct ?birthdate ?thumbnail ?author ?name ?description WHERE {
17 ?author rdf:type dbo:Writer ;
18         dbo:birthDate ?birthdate ;
19         rdfs:label ?name ;
20         dc:description ?description
21 FILTER ((lang(?name)="en")&&(lang(?description)="en")&&(STRLEN(STR(?birthdate))>6)&&(SUBSTR(STR(?birthdate),6)=SUBSTR(STR(bif:curdate('')),6))) .
22 OPTIONAL { ?author dbo:thumbnail ?thumbnail . }
23 } ORDER BY ?birthdate
24 """)
25
26 sparql.setReturnFormat(JSON)
27 results = sparql.query().convert()
28
29 # Create HTML output
30 print '<html><head><title>Literary Birthdays of Today</title></head>'
31
32 #extract Weekday %A / Month %B / Day of the Month %d by formatting today's date accordingly
33 datum = datetime.today().strftime("%A %B %d")
34 print '<body><h1>Literary Birthdays of {}</h1>'.format(datum)
35
36 print '<ul>'
37
38 for result in results["results"]["bindings"]:
39     if ("author" in result):
40         #Create Wikipedia Link
41         wikiurl = "http://en.wikipedia.org/wiki/" + result["author"]["value"].encode('ascii', 'ignore').split('/')[~1]
42     else:
43         wikiurl = 'NONE'
44     if ("name" in result):
45         name = result["name"]["value"].encode('ascii', 'ignore')
46     else:
47         name = 'NONE'
48     if ("birthdate" in result):
49         date = result["birthdate"]["value"].encode('ascii', 'ignore')
50     else:
51         date = 'NONE'
52     if ("description" in result):
```

[Link to python file](#)

...or simply use the Collaborative Python Notebook

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

 6.1 - Linked Data Programming Example.ipynb ☆

File Edit View Insert Runtime Tools Help

CODE TEXT CELL CELL

COMMENT

CONNECT

Scientific Birthday Example

[Link to collaborative Notebook](#)

This is the python notebook example for lecture 6.1 Linked Data Programming of OpenHPI "Information Service Engineering" 2018

```
[ ] !pip install -q sparqlwrapper #install SPARQLwrapper
```

We are going to use a few libraries:

- **urllib2** for URL manipulation
- **datetime** for date formatting and interpretation
- **SPARQLWrapper** to execute SPARQL queries and to import the results into python

```
[ ] import urllib2
from datetime import datetime
from SPARQLWrapper import SPARQLWrapper, JSON, XML, N3, RDF
```

We will use DBpedia as our SPARQL endpoint

```
[ ] sparql = SPARQLWrapper("http://dbpedia.org/sparql") #determine SPARQL endpoint
```

Next comes the query example from the lecture and its execution

```
[ ] #SPARQL query to be executed
sparql.setQuery("""
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

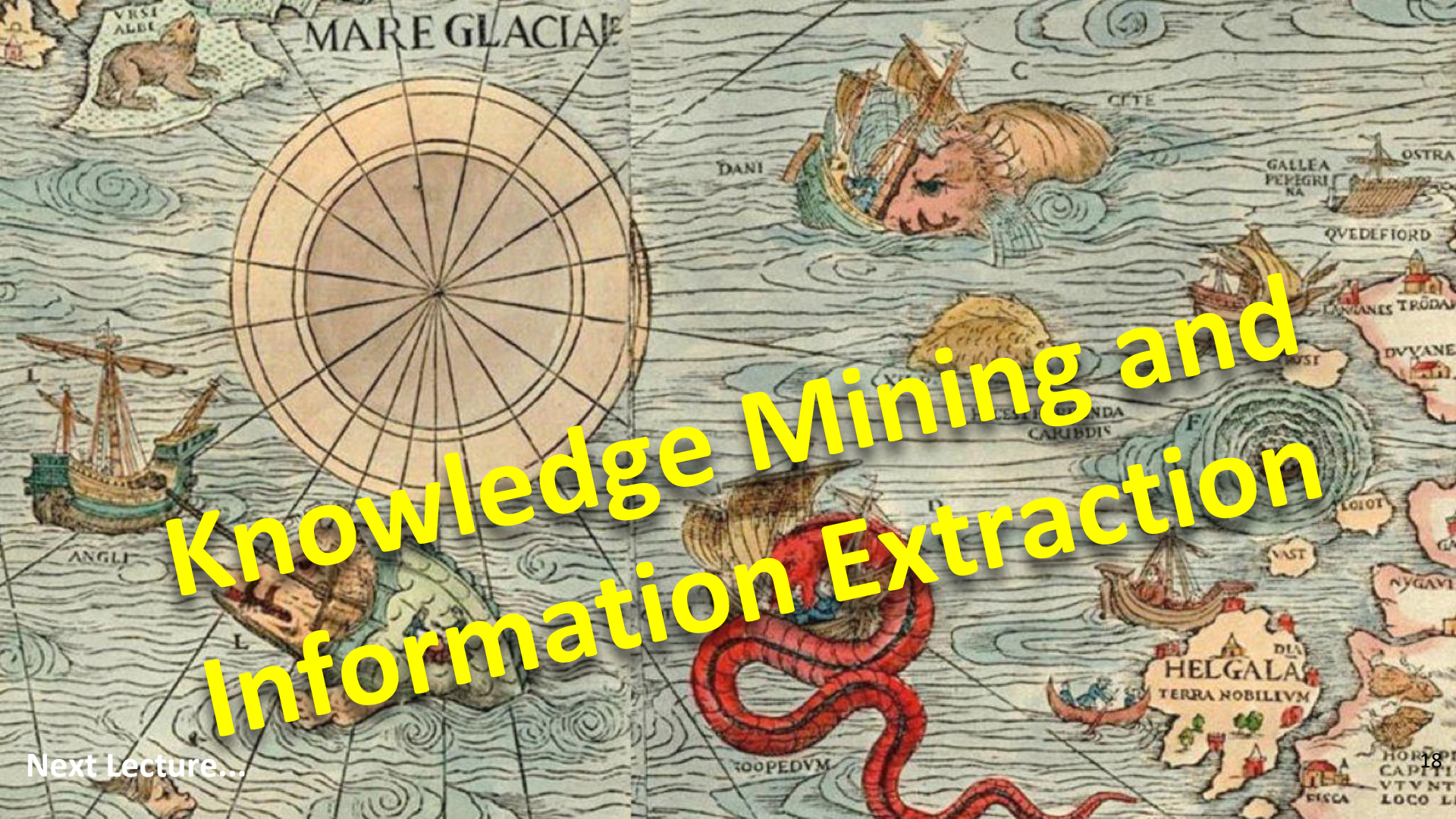
Select distinct ?birthdate ?thumbnail ?scientist ?name ?description WHERE {
?scientist rdf:type dbo:Writer ;
  dbo:birthdate ?birthdate ;
  rdfs:label ?name ;
  rdfs:comment ?description
FILTER ((lang(?name)="en")&&(lang(?description)="en")&&(STRLEN(STR(?birthdate))>6)&&(SUBSTR(STR(?birthdate),6)=SUBSTR(STR(bif:curdate('')),6))) .
OPTIONAL { ?scientist dbo:thumbnail ?thumbnail . }
""")
```

Linked Data Programming Example

Information Service Engineering - Lecture 6 Putting it All Together - 6.1 Linked Data Programming

Scientific Birthdays of Sunday April 01

- **1613** --  [Charles de Saint-vremond](#), Charles de Marguetel de Saint-Denis, seigneur de Saint-vremond (1 April 1613 29 September 1703) was a French soldier, hedonist, essayist and literary critic. After 1661, he lived in exile, mainly in England, as a consequence of his attack on French policy at the time of the Peace of the Pyrenees (1659). He is buried in Poets' Corner, Westminster. He wrote for his friends and did not intend his work to be published, although a few of his pieces were leaked in his lifetime. The first full collection of his works was published in London in 1705, after his death.
- **1874** --  [Albert Thibaudet](#), Albert Thibaudet (1 April 1874, Tournus, Sane-et-Loire 16 April 1936, Geneva) was a French essayist and literary critic. A former student of Henri Bergson, he was a professor of Jean Rousset. He taught at the University of Geneva, and was the co-founder of the Geneva School of literary criticism. He was succeeded in his post by Marcel Raymond.
- **1882** --  [Jack Cunningham \(screenwriter\)](#), Jack Cunningham (April 1, 1882 October 4, 1941) was an American screenwriter. He wrote for 132 films between 1913 and 1939. He was born in Ionia, Iowa, and died from a cerebral hemorrhage in Santa Monica, California.
- **1892** --  [Katherine Binney Shippen](#), Katherine Binney Shippen (April 1, 1892 February 20, 1980) was an American history teacher, museum curator, and children's writer.
- **1896** --  [Pola Gojawiczyska](#), Pola Gojawiczyska, real name Apolonia Gojawiczyska, ne Koniewska (1 April 1896 29 March 1963) was a Polish writer.
- **1900** --  [Tone Selikar](#), Anton "Tone" Selikar (1 April 1900 10 August 1969) was a Slovene writer, poet, journalist and teacher. He published poetry, short stories and novels and is also known for his youth literature. Together with Mile Klopi, he is considered the foremost representative of Slovene social realist poetry of the 1930s and 1940s. In 1947 he won the Preeren Award for his story Tovari (Comrades). The public library in Trbovlje is named after Selikar.
- **1901** --  [Sergiusz Piasecki](#), Sergiusz Piasecki (Polish pronunciation: [sru pjastski]; 1901 in Lachowicze near Baranowicze 1964 in Penley) - was one of the best known Polish language writers of the mid 20th century. His crowning achievement, Kochanek Wielkiej Niedwiedzicy (Lover of the Ursa Major or Lover of the Greater Bear) published in 1937, was the third most popular novel in the Second Polish Republic. Following World War II, Piasecki's books were banned by communist censorship in the People's Republic of Poland.



Knowledge Mining and Information Extraction

Next Lecture...

Picture References:

- P.18: Carta marina, a wallmap of w:Scandinavia, by Olaus Magnus.
The caption reads : Marine map and Description of the Northern Lands and of their Marvels, most carefully drawn up at Venice in the year 1539 through the generous assistance of the Most Honourable Lord Hieronymo Quirino.
https://en.wikipedia.org/wiki/File:Carta_Marina.jpeg