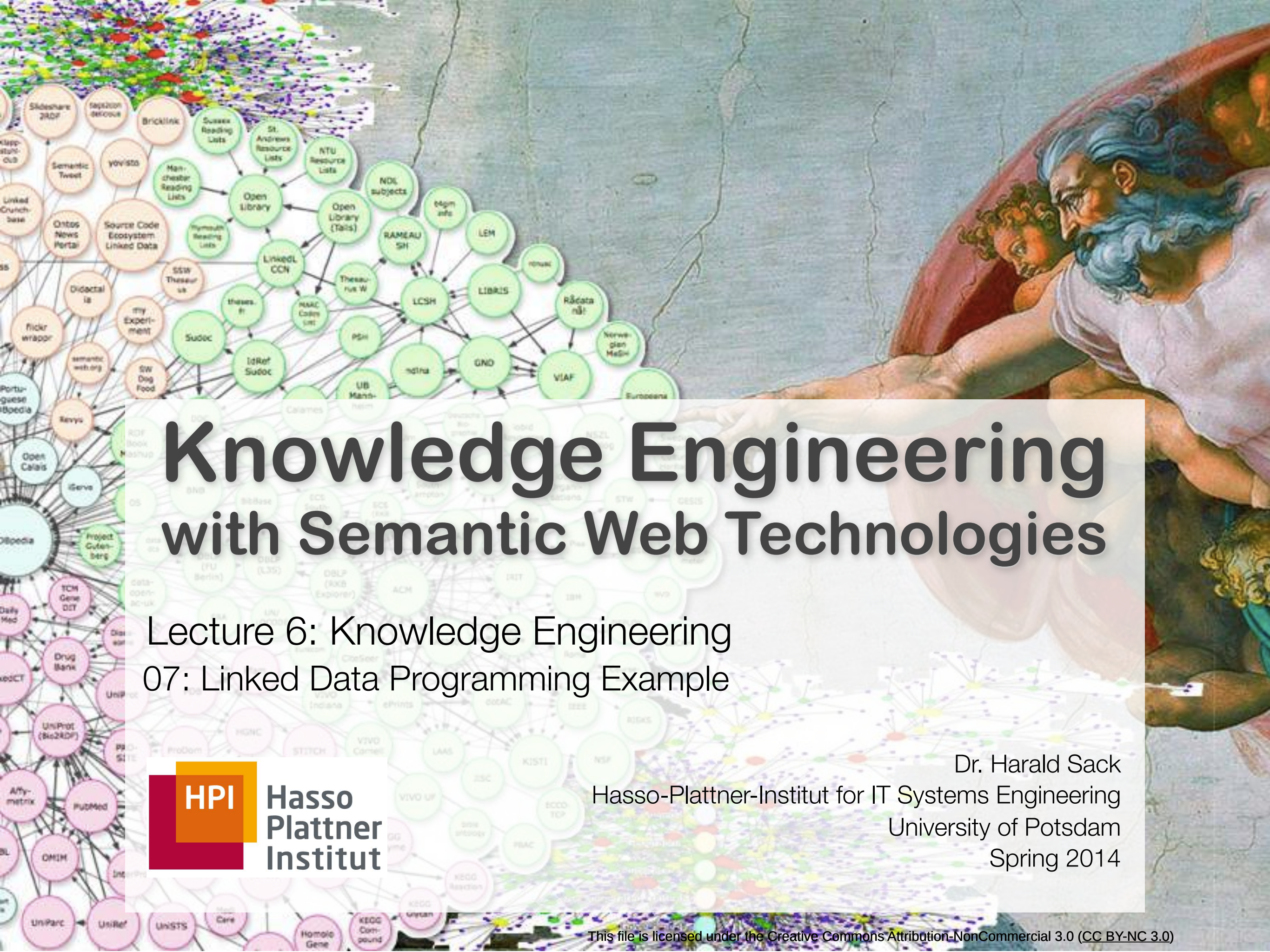




Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering
07: Linked Data Programming Example



Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

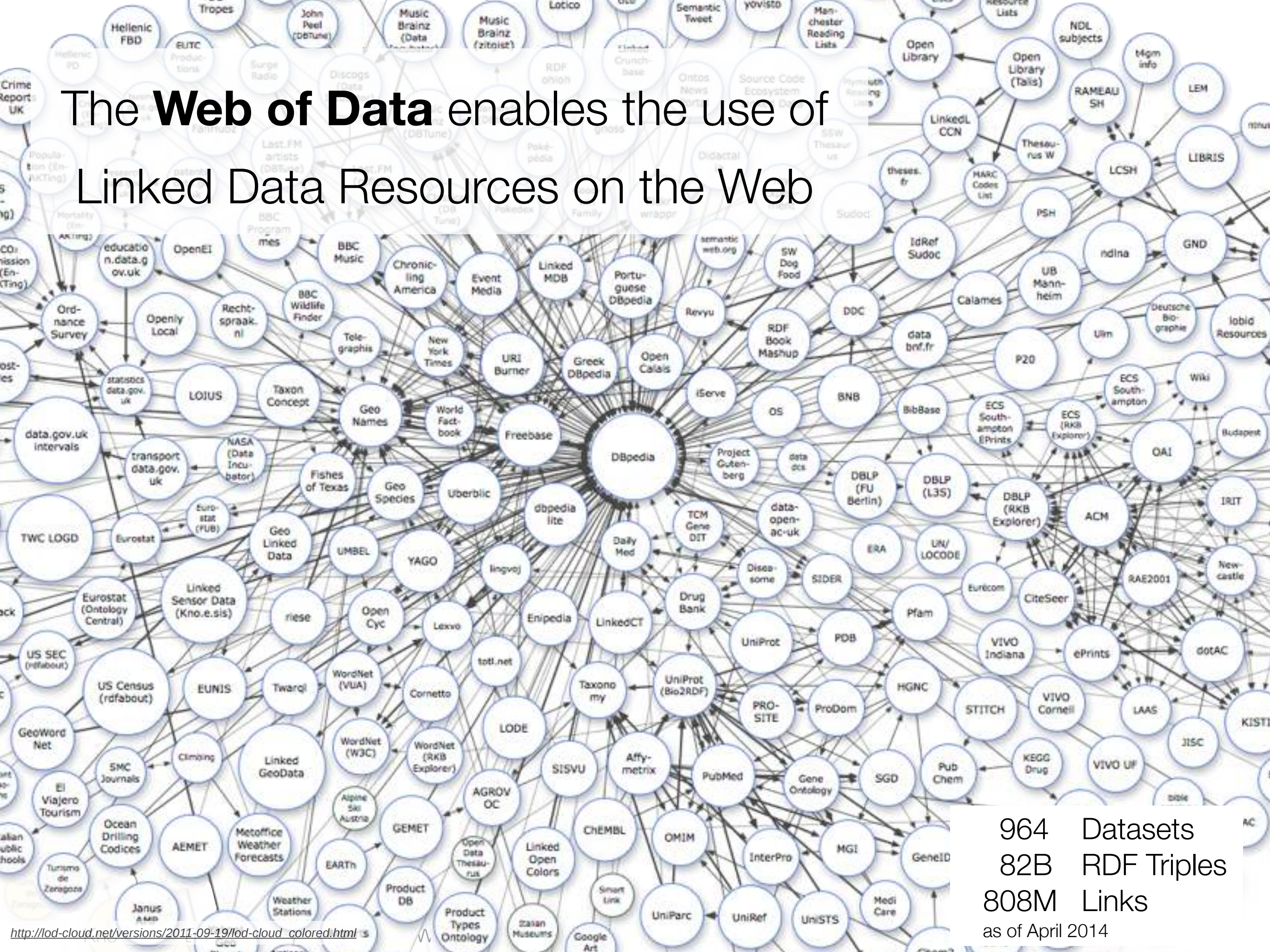


Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies



The **Web of Data** enables the use of Linked Data Resources on the Web



964 Datasets
82B RDF Triples
808M Links
as of April 2014

Linked Data Driven Web Applications

5 ☐ The easiest way is to make use of a suitable library:

☐ **SPARQL Wrapper (Python)**

☐ **<http://sparql-wrapper.sourceforge.net/>**

☐ Access to Linked Data via **SPARQL endpoints**

☐ let's choose **DBpedia** (just for simplicity...)

☐ **<http://dbpedia.org/sparql>**

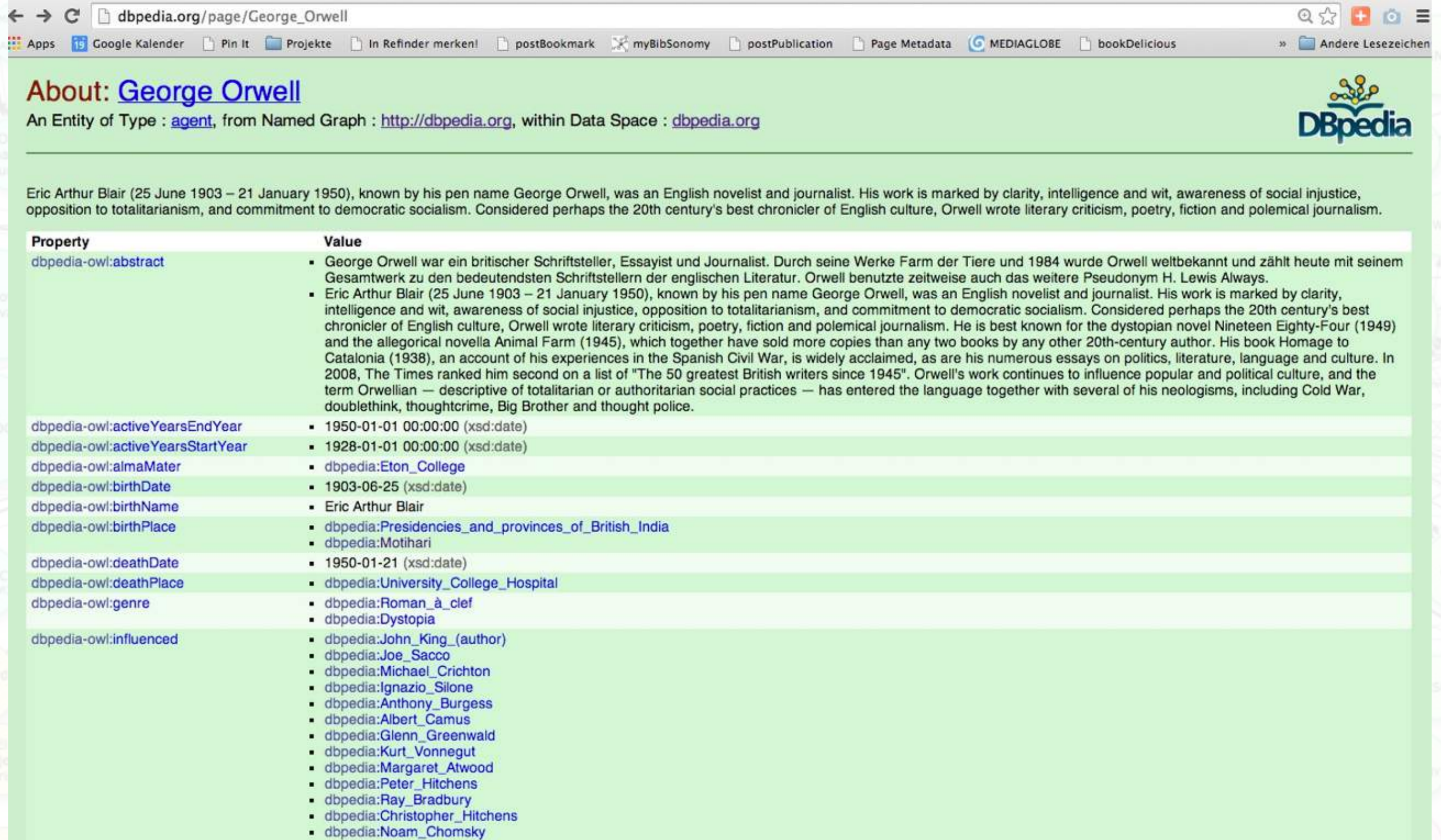
☐ ...now we have to think of a simple example...

Linked Data Programming Example

□ Example Application:

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

Linked Data Programming Example



dbpedia.org/page/George_Orwell

Apps 19 Google Kalender Pin It Projekte In Refinder merken! postBookmark myBibSonomy postPublication Page Metadata MEDIAGLOBE bookDelicious » Andere Lesezeichen

About: [George Orwell](#)

An Entity of Type : [agent](#), from Named Graph : <http://dbpedia.org>, within Data Space : [dbpedia.org](#)



Eric Arthur Blair (25 June 1903 – 21 January 1950), known by his pen name George Orwell, was an English novelist and journalist. His work is marked by clarity, intelligence and wit, awareness of social injustice, opposition to totalitarianism, and commitment to democratic socialism. Considered perhaps the 20th century's best chronicler of English culture, Orwell wrote literary criticism, poetry, fiction and polemical journalism.

Property	Value
dbpedia-owl:abstract	- George Orwell war ein britischer Schriftsteller, Essayist und Journalist. Durch seine Werke Farm der Tiere und 1984 wurde Orwell weltbekannt und zählt heute mit seinem Gesamtwerk zu den bedeutendsten Schriftstellern der englischen Literatur. Orwell benutzte zeitweise auch das weitere Pseudonym H. Lewis Always. - Eric Arthur Blair (25 June 1903 – 21 January 1950), known by his pen name George Orwell, was an English novelist and journalist. His work is marked by clarity, intelligence and wit, awareness of social injustice, opposition to totalitarianism, and commitment to democratic socialism. Considered perhaps the 20th century's best chronicler of English culture, Orwell wrote literary criticism, poetry, fiction and polemical journalism. He is best known for the dystopian novel Nineteen Eighty-Four (1949) and the allegorical novella Animal Farm (1945), which together have sold more copies than any two books by any other 20th-century author. His book Homage to Catalonia (1938), an account of his experiences in the Spanish Civil War, is widely acclaimed, as are his numerous essays on politics, literature, language and culture. In 2008, The Times ranked him second on a list of "The 50 greatest British writers since 1945". Orwell's work continues to influence popular and political culture, and the term Orwellian — descriptive of totalitarian or authoritarian social practices — has entered the language together with several of his neologisms, including Cold War, doublethink, thoughtcrime, Big Brother and thought police.
dbpedia-owl:activeYearsEndYear	1950-01-01 00:00:00 (xsd:date)
dbpedia-owl:activeYearsStartYear	1928-01-01 00:00:00 (xsd:date)
dbpedia-owl:almaMater	dbpedia:Eton_College
dbpedia-owl:birthDate	1903-06-25 (xsd:date)
dbpedia-owl:birthName	Eric Arthur Blair
dbpedia-owl:birthPlace	dbpedia:Presidencies_and_provinces_of_British_India dbpedia:Motihari
dbpedia-owl:deathDate	1950-01-21 (xsd:date)
dbpedia-owl:deathPlace	dbpedia:University_College_Hospital
dbpedia-owl:genre	dbpedia:Roman_à_clef dbpedia:Dystopia
dbpedia-owl:influenced	dbpedia:John_King_(author) dbpedia:Joe_Sacco dbpedia:Michael_Crichton dbpedia:Ignazio_Silone dbpedia:Anthony_Burgess dbpedia:Albert_Camus dbpedia:Glenn_Greenwald dbpedia:Kurt_Vonnegut dbpedia:Margaret_Atwood dbpedia:Peter_Hitchens dbpedia:Ray_Bradbury dbpedia:Christopher_Hitchens dbpedia:Noam_Chomsky

Linked Data Programming Example

Virtuoso SPARQL Query Editor

8 Default Data Set Name (Graph IRI)

Query Text

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

SELECT ?person ?name ?date ?indegree ?descr
WHERE
{
  ?person rdf:type dbpedia-owl:Writer .
  ?person dbpedia-owl:birthDate ?date FILTER (SUBSTR(STR(?date),6)=SUBSTR(STR(bif:curdate('')), 6)) .
  ?person dbpedia-owl:wikiPageInLinkCount ?indegree .
  ?person rdfs:label ?name FILTER(lang(?name)="en") .
  ?person dc:description ?descr .
}
ORDER BY DESC(?indegree)
```

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

Results Format:

HTML



(The CXML output is disabled, see [details](#))

Execution timeout:

30000

milliseconds (values less than 1000 are ignored)

Options:

☒ Strict checking of void variables

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

Run Query

Reset

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
```

```
from datetime import datetime
from SPARQLWrapper import SPARQLWrapper, JSON, XML, N3, RDF
```

```
sparql = SPARQLWrapper("http://dbpedia.org/sparql")
```

```
sparql.setQuery("""
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dctypes: <http://purl.org/dc/terms/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

SELECT ?person ?name ?date ?indegree ?descr
WHERE
{
  ?person rdf:type dbpedia-owl:Writer .
  ?person dbpedia-owl:birthDate ?date FILTER (SUBSTR(STR(?date),6)=SUBSTR(STR(bif:curdate('')), 6)) .
  ?person dbpedia-owl:wikiPageInLinkCount ?indegree .
  ?person rdfs:label ?name FILTER(lang(?name)="en") .
  ?person dc:description ?descr .
}
ORDER BY DESC(?indegree)
""")
```

```
sparql.setReturnFormat(JSON)
results = sparql.query().convert()
```

```
# Create HTML output
print '<html><head><title>Literary Birthdays of Today</title></head>'
```

```
#extract Weekday %A / Month %B / Day of the Month %d by formatting today's date accordingly
datum = datetime.today().strftime("%A %B %d")
print '<body><h1>Literary Birthdays of {}</h1>'.format(datum)
```

```
print '<ul>'
```

```
for result in results["results"]["bindings"]:
    if ("person" in result):
        #Create Wikipedia Link
        url = "http://en.wikipedia.org/wiki/" + result["person"]["value"].encode('ascii', 'ignore').split('/')[1]
    else:
        url = 'NONE'
    if ("name" in result):
        name = result["name"]["value"].encode('ascii', 'ignore')
    else:
        name = 'NONE'
    if ("date" in result):
        date = result["date"]["value"].encode('ascii', 'ignore')
    else:
        date = 'NONE'
    if ("descr" in result):
        description = result["descr"]["value"].encode('ascii', 'ignore')
    else:
        description = ' '
```

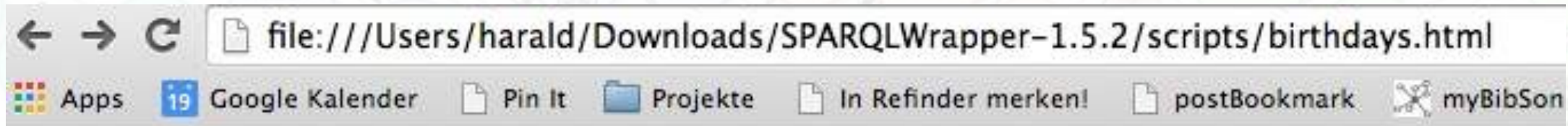
```
#parse date as datetime
dt = datetime.strptime(date, '%Y-%m-%d+02:00')
print '<li><b>{}</b> -- <a href={}>{}</a>, {} </li>'.format(dt.year, url, name, description)
```

```
print '</ul>'
```

```
print '</body></html>|'
```



Linked Data Programming Example



Literary Birthdays of Thursday June 19

- 1947 -- [Salman Rushdie](#), British-Indian novelist and author
- 1572 -- [John Donne](#), English poet
- 1945 -- [Tobias Wolff](#), Fiction writer, memoirist
- 1909 -- [Osamu Dazai](#), Japanese author
- 1928 -- [Barry Took](#), British comedian
- 1952 -- [Angie Sage](#), English author and illustrator
- 1921 -- [Patricia Wrightson](#), Australian children's writer
- 1950 -- [Rosie Shuster](#), Canadian-born comedy writer
- 1932 -- [Mark Adlard](#), Novelist
- 1969 -- [Thomas Breitling](#), Entrepreneur
- 1909 -- [Henryka azowertwna](#), Polish writer
- 1872 -- [Sutton E. Griggs](#), Novelist, clergyman, social activist
- 1988 -- [Sarah Kay \(poet\)](#), American poet
- 1975 -- [Rekha Waheed](#), Novelist
- 1912 -- [Luu Trong Lu](#), Poet
- 1969 -- [Hlne Boudreau](#), Canadian children's author
- 1891 -- [Carlos Wyld Ospina](#), Guatemalan writer
- 1980 -- [Sadaf Fayyaz](#), Journalist

Linked Data Driven Web Applications

- Alternative: simple example with Jena ARQ and Java:

```
import com.hp.hpl.jena.query.*;

String service = "..."; // address of the SPARQL endpoint
String query = "SELECT ..."; // your SPARQL query
QueryExecution e = QueryExecutionFactory.sparqlService(service, query)

ResultSet results = e.execSelect();
while ( results.hasNext() ) {
    QuerySolution s = results.nextSolution();
    // ...
}

e.close();
```

- <http://jena.sourceforge.net/>



08 - More Applications

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering