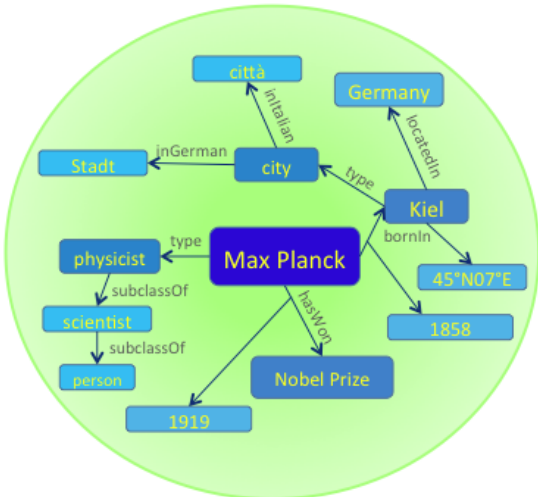
# Wikipedia in Triples

(you can use this for your project)

is a huge semantic knowledge base, which contains
structured information about Wikipedia entities
(people, cities, locations, etc.)

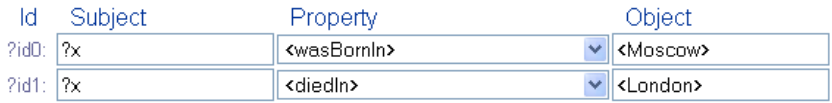
https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/



# Using Yago

Find famous people who were born in Moscow and died in London

by querying Yago at https://gate.d5.mpi-inf.mpg.de/webyagospotlx/WebInterface




You can also execute the SPARQL query

```
PREFIX ya: <http://yago-knowledge.org/resource/>
SELECT ?x
WHERE
{
  ?x  ya:wasBornIn  ya:Moscow  .
  ?x  ya:diedIn  ya:London  .
}
```

at http://lod2.openlinksw.com/sparql

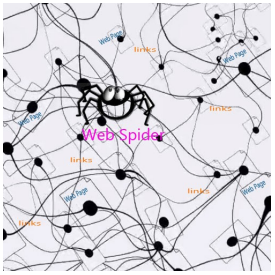
## How would you build such a knowledge base?

Yago was constructed using **infoboxes** and **categories** from Wikipedia

see https://suchanek.name/work/publications/submitted.pdf

Our aim today is to extract from the Wikipedia pages RDF triples about American authors

Tools:




– **web crawler** (Internet bot which browses the Web)

– **regular expressions** (sequences of symbols expressing a string or pattern to be searched for within a longer piece of text)

[A-Z][a-z]* any word that starts with a capital character

([A-Z][a-z]* )+ any sequence of one or more such words

[1-9][0-9][0-9][0-9] year (you can also use \d\d\d\d)

[^ >] any character different from >

(good regular expression cheat sheet http://krijnhoetmer.nl/stuff/regex/cheat-sheet)

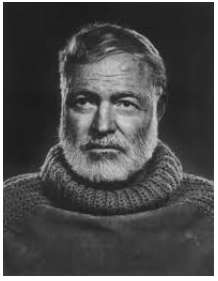
# Crawling Wikipedia

Use the `wget` web crawler to download a piece of Wikipedia on your computer:

```
wget --random-wait -r -l3 -p -e robots=off
                                    en.wikipedia.org/wiki/Ernest_Hemingway
```

with `en.wikipedia.org/wiki/Ernest_Hemingway` as a starting point

Ernest Miller Hemingway (July 21, 1899 – July 2, 1961)



you can also download and unzip `http://www.dcs.bbk.ac.uk/~kikot/wiki.zip`

# Exercise

1. Write a command that outputs **names** of people together with their **birthplaces** from Wikipedia in the TURTLE format

```
name1  :wasBornIn  place1  .
name2  :wasBornIn  place2  .
...
```

2. Write a command that outputs **names and dates of birth** in the format

```
name1  :wasBornOn  date1  .
name2  :wasBornOn  date2  .
...
```

# Hint

Use Linux tools such as grep, egrep, awk, sed, perl etc.

```
cat * | grep 'was born'
```

**cat *** | sends all the files to the standard input of the command to follow

**grep** outputs only those lines that contain a given phrase

cat * | egrep `([A-Z][a-z]* )+was born'

**egrep** outputs only those lines that match a given regular expression)

Use **perl** for extraction of separate matches

cat * | perl -ne 'if ($_ =~ m/(([A-Z][a-z]* )+)was born /) {print $_."\n"} ;'

**perl -ne** applies the `...' command to every input line (denoted $_) )

cat * | perl -ne 'if ($_ =~m/(($2([A-Z][a-z]* )$1 +)was born in/) {print $1."\n"} ;'

# Answers

1) `cat * | perl -ne 'if ($_ =~ m/(([A-Z][a-z]* )+)was born in <a[^>]*>(.*?)</ ) {print $1." :wasBornIn ".$3."\n"}';`

(place of birth usually lies just between <a . . . > and </a> tags )

2) cat * | perl -ne 'if ($_ =~ m/(([A-Z][a-z]* )+)was born on (.*?\d\d\d\d)/) {print $1." :wasBornOn ".$3."\n"} ;'

(date of birth ends with a number, which consists of 4 digits (\d) )