Discovering Knowledge in Linked Data

James Earl Douglas

Text by the Bay 2015

The approach

- Explore individual facts
- Connect related data
- Synthesize the bigger picture

Explore individual facts

- Richard Feynman was born in Queens
- Queens is a borough of New York City
- New York City is in the United States

Connect related data

- Queens is a borough New York City
- New York City is in the United States

Queens is in the United States.

Synthesize the bigger picture

- Richard Feynman was born in Queens
- Queens is in the United States

Richard Feynman was born in Queens, which is a borough of New York City in the United States.

How to get there

Build on foundations of the Semantic Web.

Semantic Wuzzah?

Semantic Web

It's the Web we all know, but with a bit of structure around the information.

Where is it?

Lots of places!

- Wikidata, DBpedia, Freebase
- Data.gov
- MusicBrainz
- The actual Web

What's structure?

Consider an HTML list.

Rather than a flat string, "*electrons, protons, and neutrons*", it has structure:

```
<ul>
  <li>Electrons</li>
  <li>Protons</li>
  <li>Neutrons</li>
</ul>
```

Moar structure: triples

Richard Feynman plays the bongo drum.

- Subject: the primary resource being described
- Predicate: the releation between subject and object
- Object: the value of the relation

Resource Description Framework

An ecosystem of standards for specifying, among other things, triples.

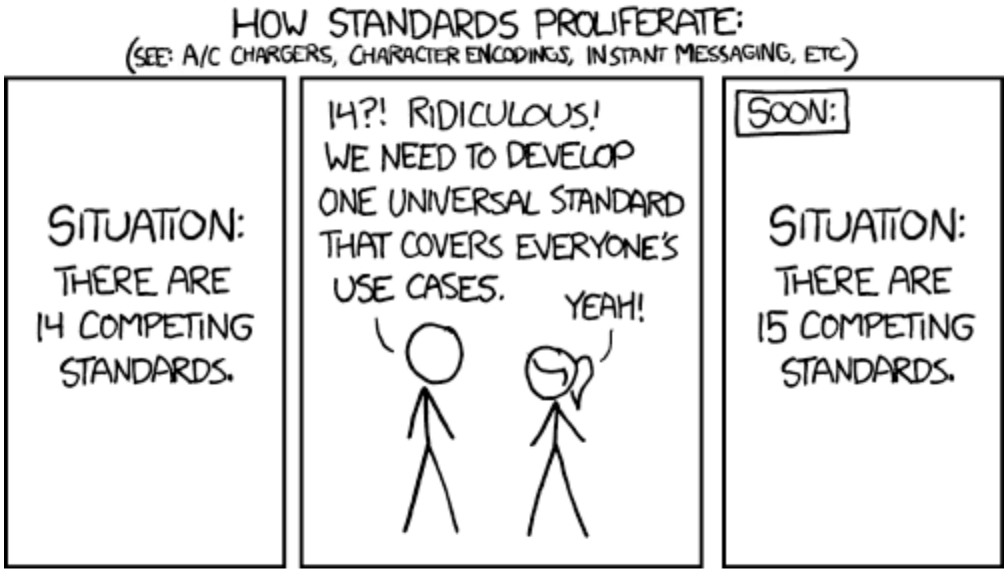
- `<http://www.wikidata.org/entity/Q39246>`
- `<http://www.wikidata.org/entity/P1303>`
- `<http://www.wikidata.org/entity/Q243998>`

Trouble with Triples

Triples are great at capturing facts, but seem overwhelmingly complex.

Too many standards

RDF, RDFS, N-Triples, Turtle, SPARQL, etc.


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

https://xkcd.com/927/

Unreadability of RDF

What does this say?

```
<http://www.wikidata.org/entity/Q39246>
  <http://www.wikidata.org/entity/P108s>
  <http://www.wikidata.org/entity/Q39246SE1E55ECD-9A13-49BC-B6FD-99995E4C0FC7> .
<http://www.wikidata.org/entity/Q39246SE1E55ECD-9A13-49BC-B6FD-99995E4C0FC7>
  <http://www.wikidata.org/entity/P108v>
  <http://www.wikidata.org/entity/Q49115> .
<http://www.wikidata.org/entity/Q49115>
  <http://www.wikidata.org/entity/P373s>
  <http://www.wikidata.org/entity/Q49115SEE3BD997-DA49-4A40-8648-2118D414D82D> .
<http://www.wikidata.org/entity/Q49115SEE3BD997-DA49-4A40-8648-2118D414D82D>
  <http://www.wikidata.org/entity/P373v>
  "Cornell University" .
```

"*Richard Feynman works for Cornell University.*"

Turtle makes it better

De-duplicate some of the redundancy.

```
@prefix entity: <http://www.wikidata.org/entity/> .

entity:Q39246
  entity:P108s
  entity:Q39246SE1E55ECD-9A13-49BC-B6FD-99995E4C0FC7 .
entity:Q39246SE1E55ECD-9A13-49BC-B6FD-99995E4C0FC7
  entity:P108v
  entity:Q49115 .
entity:Q49115
  entity:P373s
  entity:Q49115SEE3BD997-DA49-4A40-8648-2118D414D82D .
entity:Q49115SEE3BD997-DA49-4A40-8648-2118D414D82D
  entity:P373v
  "Cornell University" .
```

SPARQL does too

```
PREFIX entity: <http://www.wikidata.org/entity/>

SELECT ?employer WHERE {
  entity:Q39246 entity:P108s ?a .
  ?a entity:P108v ?b .
  ?b entity:P373s ?c .
  ?c entity:P373v ?employer .
}
```

Moreso with property paths

```
PREFIX entity: <http://www.wikidata.org/entity/>

SELECT ?employer WHERE {
  entity:Q39246
    entity:P108s/entity:P108v/entity:P373s/entity:P373v
    ?employer .
}
```

Domain-specific language

Let's extend SPARQL just a little bit, to make things even simpler.

- `[Q:]` namespace for entities
- `[P:]` namespace for properties
- `[O:]` namespace for ontology
- `[X:]` namespace for XSD

So fresh, so clean

```
SELECT ?employer WHERE {
  [Q:feynman] [P:employedBy]/[P:labelled] ?employer .
}
```

What can we do with it?

Ask simple questions, such as "What happened on this day in history?".

```
SELECT ?entity ?date WHERE {
  ?entityS ?x           ?dateS        .
  ?dateS   ?y           ?dateV        .
  ?dateV   [O:calendar] [Q:gregorian] .
  ?dateV   [O:time]     ?date         .
  ?entityS [P:labelled] ?entity       .
  FILTER ( regex(str(?date), "\\d{4}-\\d{2}-\\d{2}") )
  FILTER ( [X:int](substr(str(?date), 6, 2)) = month(now()) )
  FILTER ( [X:int](substr(str(?date), 9, 2)) = day(now()) )
}
```

What happened on this day in history?

entity	date
Mexican-American War	1846-04-24
Girolamo Crescentini	1846-04-24
Philatelic fakes and forgeries	1846-04-24
David Oliver	1982-04-24
Kelly Clarkson	1982-04-24
John M. Ashbrook	1982-04-24

What else can we do with it?

Ask tricky questions, such as "What were some of the fields of work of physicists who worked at institutions where Richard Feynman also worked?".

```
SELECT ?colleague ?field ?employer WHERE {
  [Q:feynman]  [P:employedBy]   ?employerS    .
  ?colleagueS  [P:employedBy]   ?employerS    .
  ?colleagueS  [P:occupiedAs]   [Q:physicist] .
  ?employerS   [P:labelled]     ?employer     .
  ?colleagueS  [P:labelled]     ?colleague    .
  ?colleagueS  [P:worksInField] ?fieldS       .
  ?fieldS      [P:labelled]     ?field        .
}
```

What is the answer to that wicked long question?

colleague	field	employer
Richard Feynman	Particle physics	Cornell University
Richard Feynman	Particle physics	California Institute of Technology
J. Robert Oppenheimer	Theoretical physics	California Institute of Technology
J. Robert Oppenheimer	Nuclear physics	California Institute of Technology

Explore statements, acquire knowledge

The Semantic Web may seem daunting at first, but it's worth the trouble.

Following connections wrapped up in related statements, we can build an enormous map of increasingly complex understanding.

- Learn things we didn't realize
- Discover relevances we didn't expect

Get involved

- Wikidata *(wikidata.org)*
- Wikidata Query Service *(mediawiki.org)*
- Blazegraph *(blazegraph.com)*

References and further reading

- SPARQL Query Language for RDF
- RDF Schema 1.1
- Wikidata
- Wikidata RDF exports
- Wikidata Query Service
- Blazegraph