OWL vs. Linked Data: Experiences and Directions

Axel Polleres
Siemens AG Österreich

27/05/2013

Disclaimer: this is not the panel you were looking for

> **Axel Polleres** shared a link.
> 4 hours ago
>
> oh my, why did I agree to give a standup invited talk to fill up for the planned panel at OWLED2013 tomorrow? http://webont.org/owled/2013/schedule.html Preparing slides in the hotel room...drop by, if you're there!

Will try to touch upon the following questions:

- Which parts of OWL are used within Linked Data?
- Which parts of OWL2 could be useful for Linked Data ?
- Which reasoning techniques can be applied to handle the scale, messiness, and dynamicity of Linked Data?
- Which reasoning beyond RDFS and OWL is necessary for Linked Data?

I don't have complete solutions to all these questions, but some on how OWL can be applied to Linked Data collected over the past view years.

Special thanks to my co-authors: **Aidan Hogan, Jürgen Umbrich, Stefan Bischof, Andreas Harth, Birte Glimm, Markus Krötzsch**, etc.

Let’s start from the beginning…

“If ***HTML*** *and the* ***Web*** *made all the online documents look like one huge* ***book****,*
RDF*,* ***schema*** *and* ***inference*** *languages will make all the data in the world look like*
one huge ***database****”* *Tim Berners-Lee, Weaving the Web, 1999*

Great! We have 2013, this should work by now, shouldn‘t it? Let‘s try that on google!

Scenario: *“Market research” on the Web, (no FOAF today, sorry)*

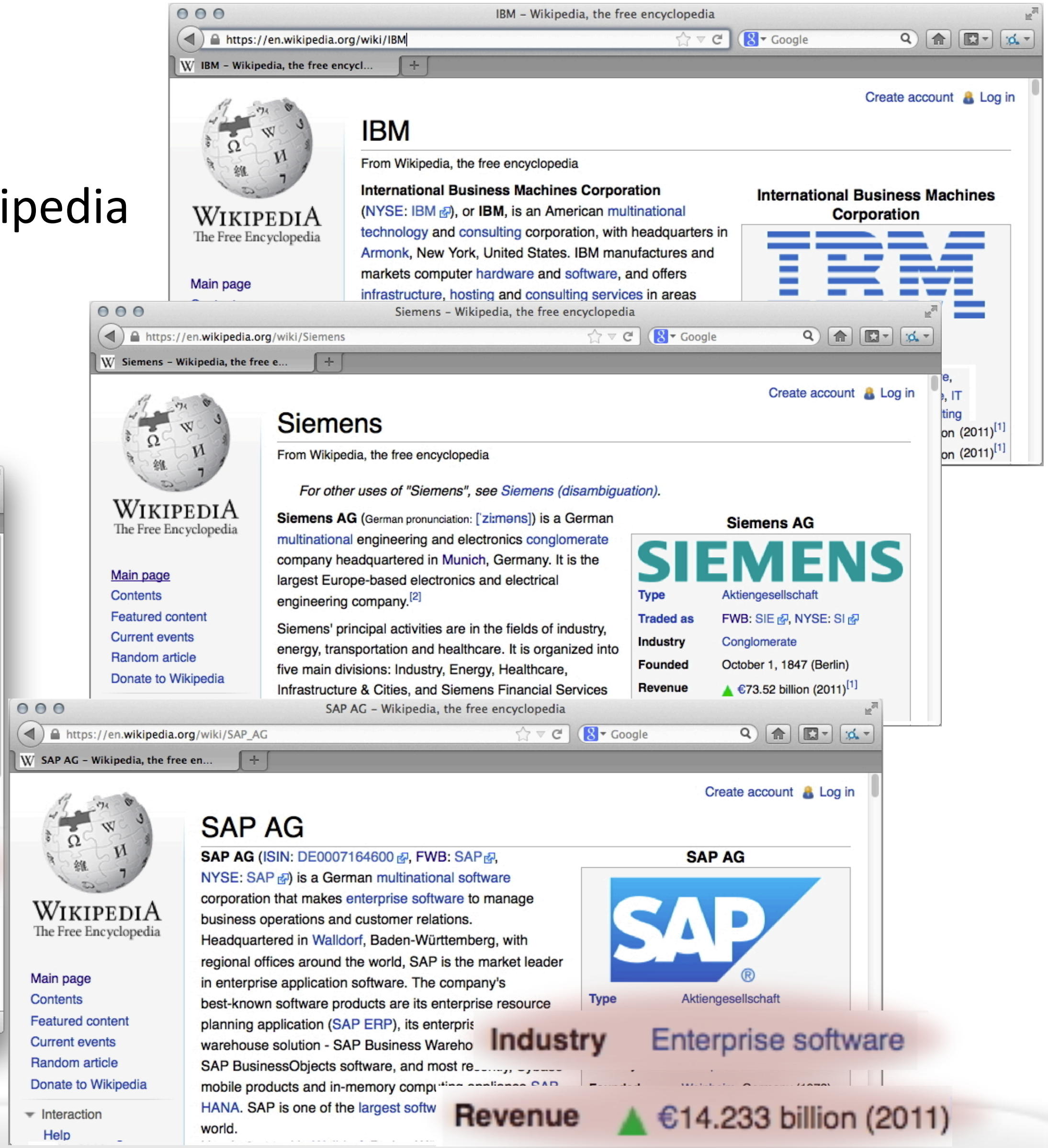
Google “Latest news on NYT about technology companies with a revenue greater than 10B EUR”

Google “Which city of Montpellier and Vienna has the higher population density?”

"Latest News on NYT about technology companies with a revenue greater than 10B EUR"

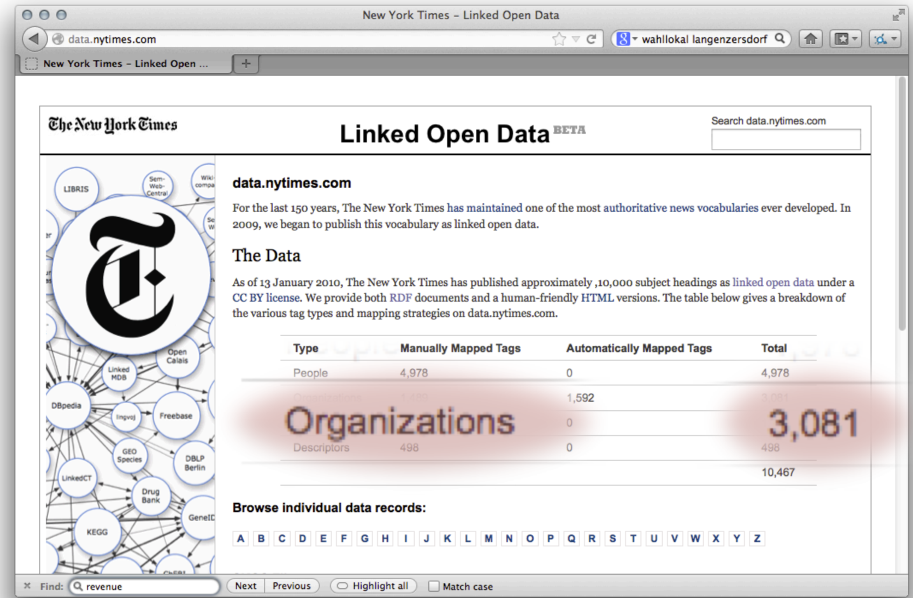
- The data is there!

Wikipedia



New York Times thesaurus

and article search API

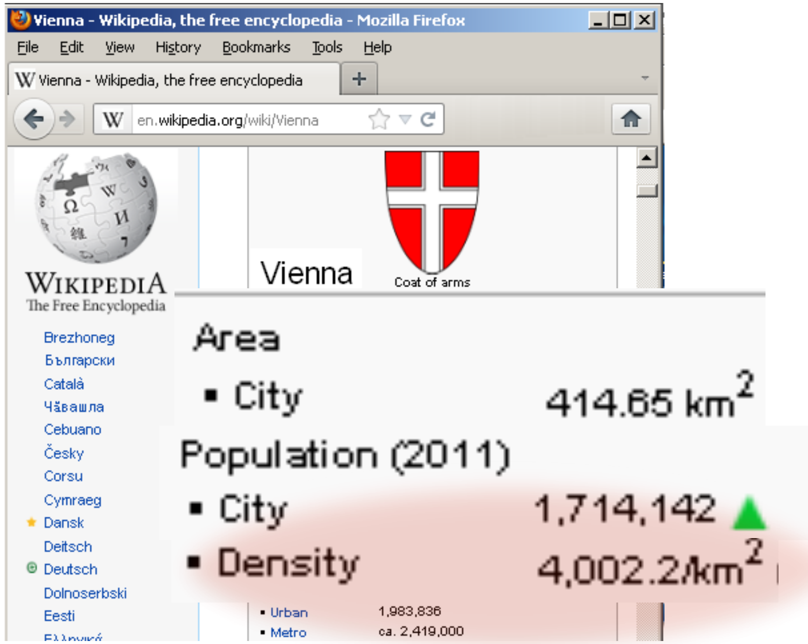


"Which city of Montpellier and Vienna has the higher population density?"

- Again:

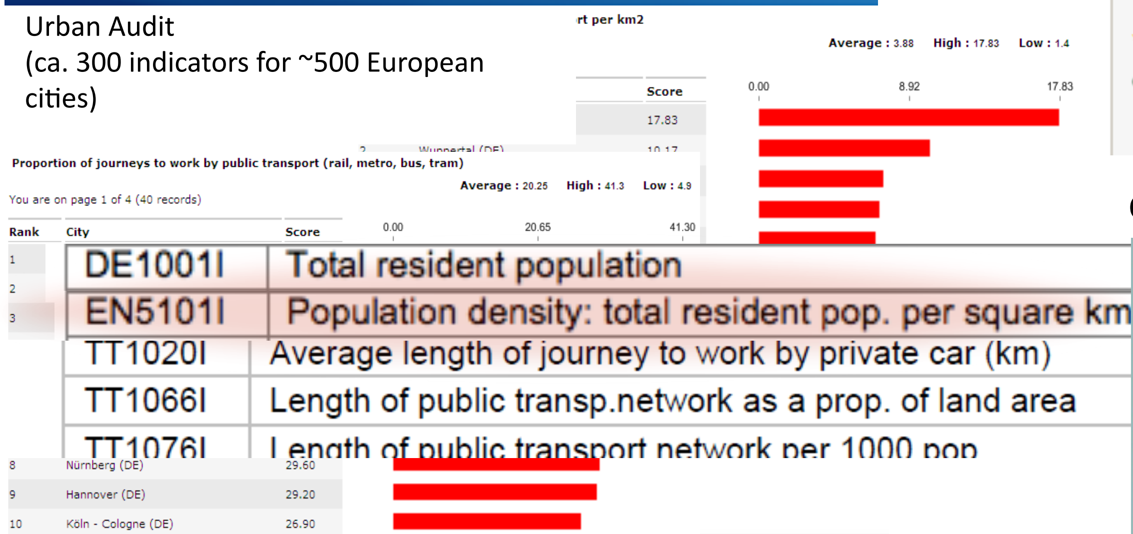
  The data is there!

Wikipedia



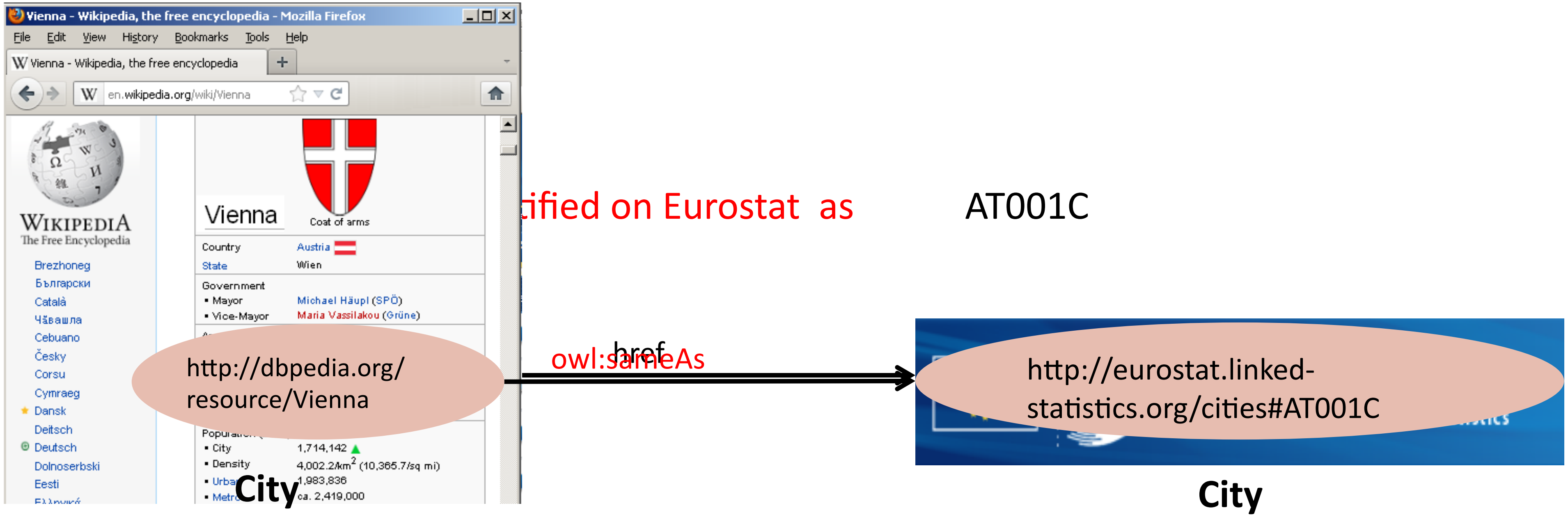
Eurostat

Urban Audit
(ca. 300 indicators for ~500 European cities)



or also inside Siemens!

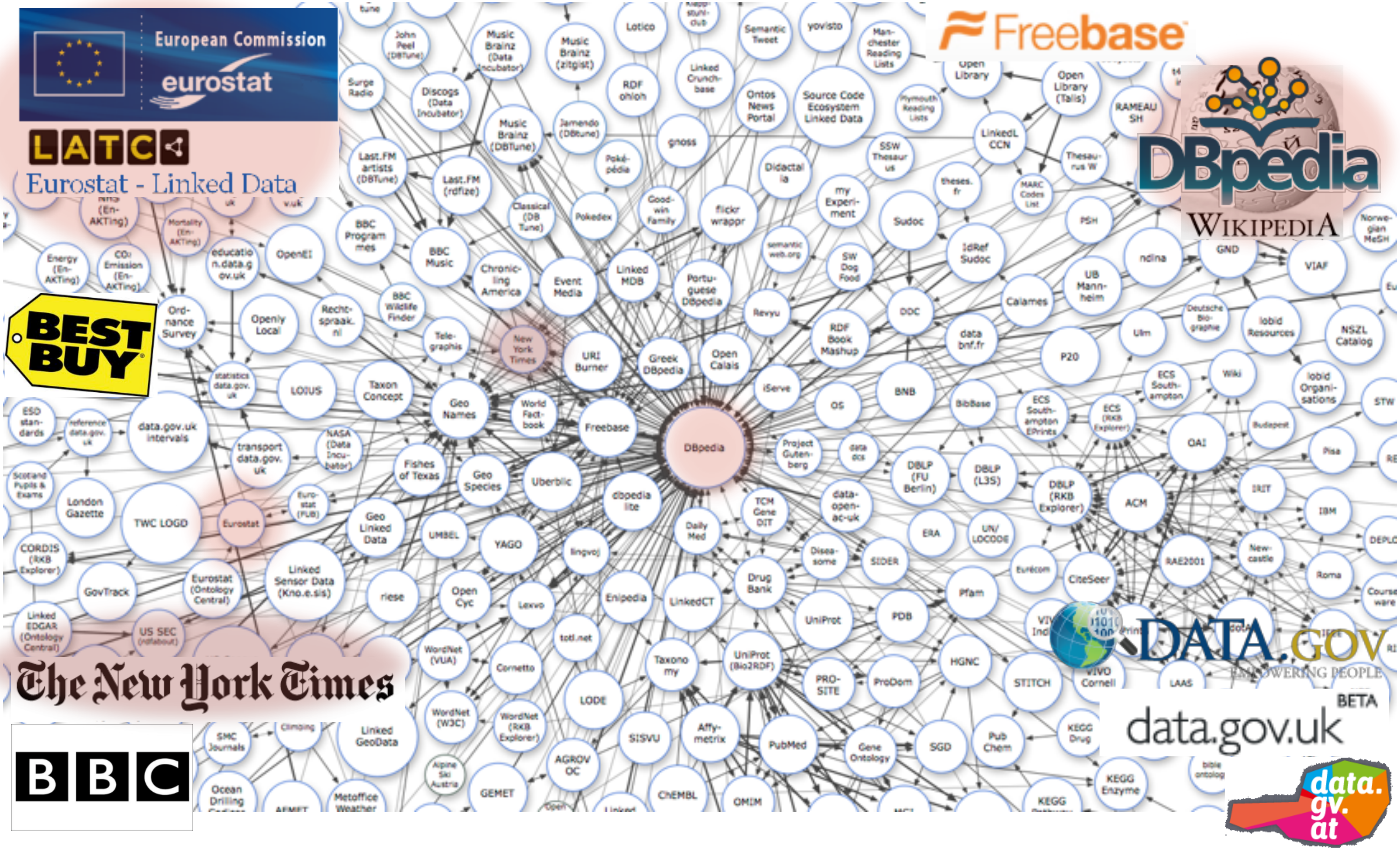
Emerging trend: Linked data



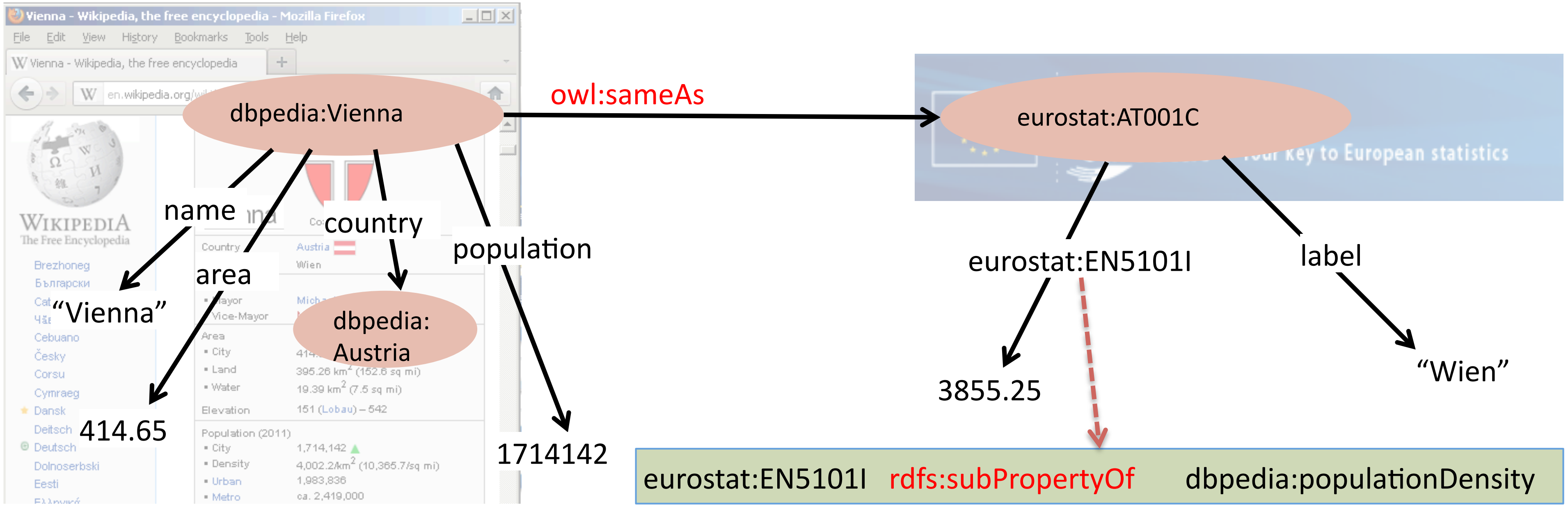
1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards (RDF, OWL)
4. Include links to other URIs. so that they can discover more things.

http://www.w3.org/DesignIssues/LinkedData.html

Linked Data on the Web: Adoption



Example: Using OWL and SPARQL to query Linked data



"Which city of Montpellier and Vienna has the higher population density?"

```
SELECT ?DM ?DV
WHERE { dbpedia:Montpellier :populationDensity ?DM .
        dbpedia:Vienna :populationDensity ?DV .
      }
```

The Semantic promise…

*"If **HTML** and the Web made all the online documents look like one huge **book**, **RDF**, **schema** and **inference** languages will make all the data in the world look like one huge **database**"* *Tim Berners-Lee, Weaving the Web, 1999*

Great! We have 2013, this should work by now, shouldn't it? Let's try that on google!

Scenario: *"Market research" on the Web,*

Google "Latest news on NYT about technology companies with a revenue greater than 10B EUR"

```
SELECT * WHERE
{ ?C rdf:type NYT:Org .
  ?C dbpedia:revenue ?R .
  ?C NYT:latestArticle ?A .
  FILTER( ?R > 10000000000 )  }
```

Google

The Semantic promise…

*"If **HTML** and the Web made all the online documents look like one huge **book**, **RDF**, **schema** and **inference** languages will make all the data in the world look like one huge **database**"* *Tim Berners-Lee, Weaving the Web, 1999*

Great! We have 2013, this should work by now, shouldn't it? Let's try that on google!

Scenario: *"Market research" on the Web,*

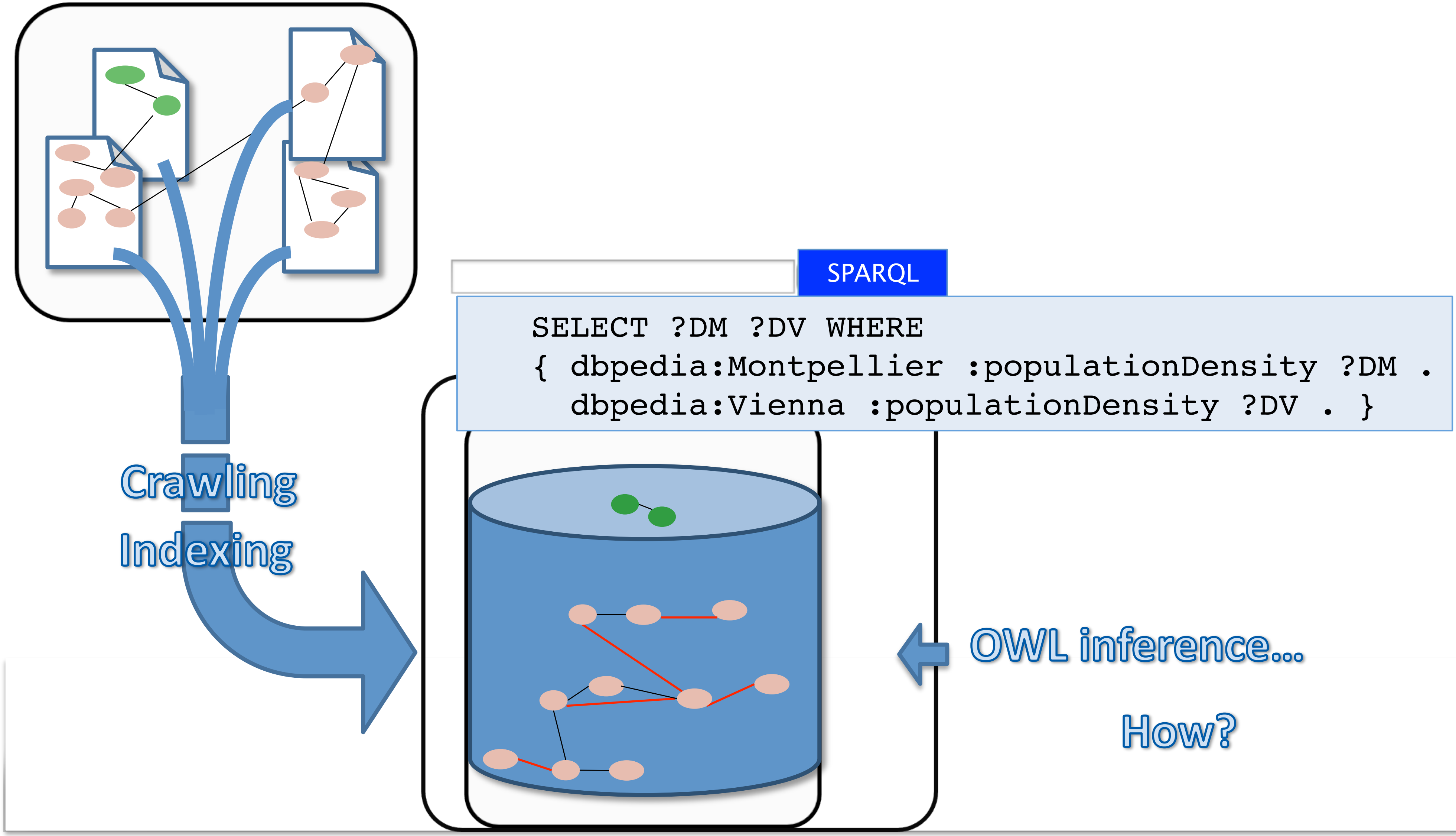
Google: "Latest news on NYT about technology companies with a revenue greater than 10B EUR"

Google: "Cities in France with a higher population density than Montpellier?"

```
SELECT ?DM ?DV
WHERE { dbpedia:Montpellier :populationDensity ?DM .
        dbpedia:Vienna :populationDensity ?DV .
      }
```

So, all we need to do:

- … is to through together a crawler, an RDF Store, some OWL inference and a SPARQL engine? Unfortunately it's not that easy…



Obstacles/prejudices:

- OWL is too hard to learn?

- OWL is too expensive?

Linked Data publishers…

www.w3.org/TR/2009/REC-owl2-direct-semantics-20091027/

C Recommendation

2.2 Interpretations

Given a datatype map D and a vocabulary V over D, an *interpretation* $I = (\Delta_I , \Delta_D , \cdot^C , \cdot^{OP} , \cdot^{DP} , \cdot^I , \cdot^{DT} , \cdot^{LT} , \cdot^{FA})$ for D and V is a 9-tuple with the following structure:

- Δ_I is a nonempty set called the *object domain*.
- Δ_D is a nonempty set disjoint with Δ_I called the *data domain* such that $(DT)^{DT} \subseteq \Delta_D$ for each datatype $DT \in V_{DT}$.
- $\cdot^C$ is the *class interpretation function* that assigns to each class $C \in V_C$ a subset $(C)^C \subseteq \Delta_I$ such that
  - $(owl:Thing)^C = \Delta_I$ and
  - $(owl:Nothing)^C = \emptyset$.
- $\cdot^{OP}$ is the *object property interpretation function* that assigns to each object property $OP \in V_{OP}$ a subset $(OP)^{OP} \subseteq \Delta_I \times \Delta_I$ such that
  - $(owl:topObjectProperty)^{OP} = \Delta_I \times \Delta_I$ and
  - $(owl:bottomObjectProperty)^{OP} = \emptyset$.
- $\cdot^{DP}$ is the *data property interpretation function* that assigns to each data property $DP \in V_{DP}$ a subset $(DP)^{DP} \subseteq \Delta_I \times \Delta_D$ such that
  - $(owl:topDataProperty)^{DP} = \Delta_I \times \Delta_D$ and
  - $(owl:bottomDataProperty)^{DP} = \emptyset$.
- $\cdot^I$ is the *individual interpretation function* that assigns to each individual $a \in V_I$ an element $(a)^I \in \Delta_I$.
- $\cdot^{DT}$ is the *datatype interpretation function* that assigns to each datatype $DT \in V_{DT}$ a subset $(DT)^{DT} \subseteq \Delta_D$ such that
  - $\cdot^{DT}$ is the same as in D for each datatype $DT \in N_{DT}$, and
  - $(rdfs:Literal)^{DT} = \Delta_D$.
- $\cdot^{LT}$ is the *literal interpretation function* that is defined as $(lt)^{LT} = (LV , DT)^{LS}$ for each $lt \in V_{LT}$, where LV is the lexical form of lt and DT is the datatype of lt.
- $\cdot^{FA}$ is the *facet interpretation function* that is defined as $(F , lt)^{FA} = (F , (lt)^{LT})^{FS}$ for each $(F , lt) \in V_{FA}$.

The following sections define the extensions of $\cdot^{OP}$, $\cdot^{DT}$, and $\cdot^C$ to object property expressions, data ranges, and class expressions.

2.2.1 Object Property Expressions

The object property interpretation function $\cdot^{OP}$ is extended to object property expressions as shown in Table 1.

…OWL IS HARD

(…to learn, to understand, to implement, to compute, to teach, to represent in RDF, to publish, to parse, to use *appropriately*…)

Corollary1:	Linked Data publishers only use a little bit of OWL …
Corollary2:	… they still manage to make mistakes ☺

Corollary1: Linked Data publishers only use a little bit of OWL …

Proof

How much OWL is used in Linked Data? [LDOW'12]

- Looked at Billion Triple Challenge 2011 Dataset (BTC2011)
  - 2.1 billion quadruples, crawled from…
  - 7.4 million RDF/XML documents, covering…
  - 791 (pay-level) domains
- Count OWL features used in the dataset:
  - Per use
  - Per document
  - Per domain
  - Can be skewed by data
- Ranked OWL features using PageRank:
  - Rank documents based on dereferenceable links
  - For each OWL feature, sum the rank of documents using it
  - **Intuition: Approximates probability of encountering an OWL feature during a random walk of the data**

Results of ranking (see LDOW'12 paper details)

1	**rdf:Property**	5.74E-1
2	**rdfs:range**	4.67E-1
3	**rdfs:domain**	4.62E-1
4	**rdfs:subClassOf**	4.60E-1
5	**rdfs:Class**	4.45E-1
6	**rdfs:subPropertyOf**	2.35E-1
7	**owl:Class**	1.74E-1
8	**owl:ObjectProperty**	1.68E-1
9	**rdfs:Datatype**	1.68E-1
10	**owl:DatatypeProperty**	1.65E-1
11	**owl:AnnotationProperty**	1.60E-1
12	**owl:FunctionalProperty**	9.18E-2
13	**owl:equivalentProperty**	8.54E-2
14	**owl:inverseOf**	7.91E-2
15	**owl:disjointWith**	7.65E-2

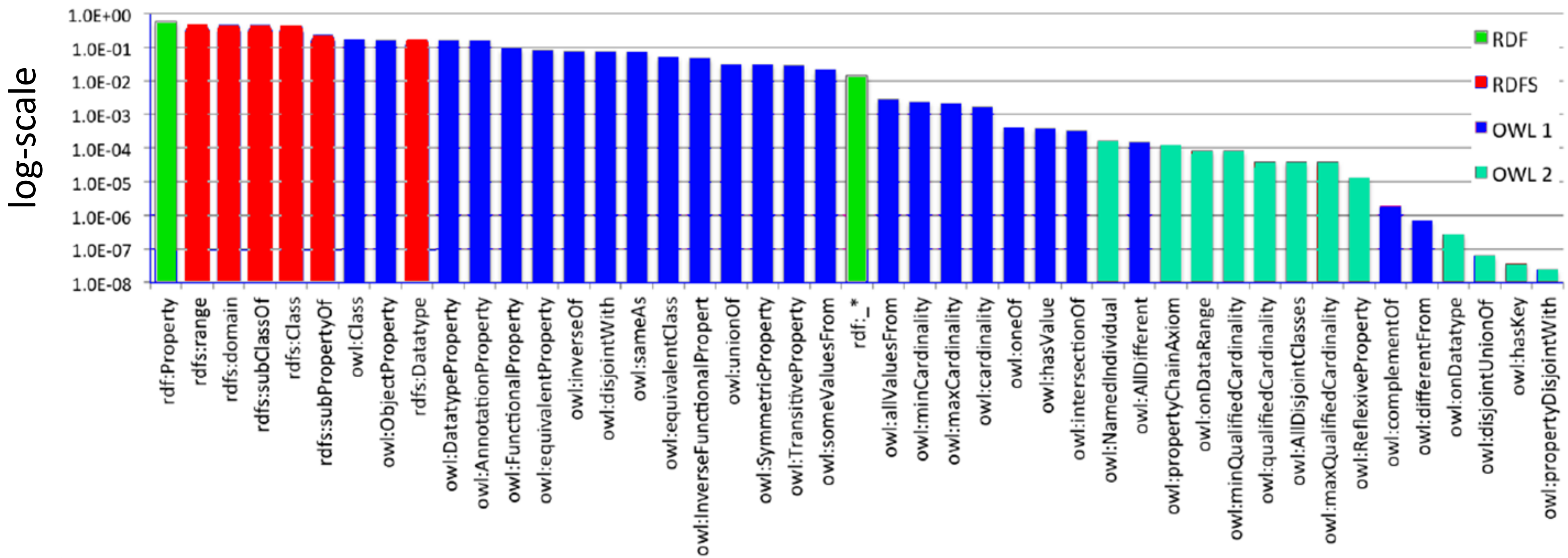
Results of ranking (see LDOW'12 paper details)

…		
16	**owl:sameAs**	7.29E-2
17	**owl:equivalentClass**	5.24E-2
18	**owl:InverseFunctionalProperty**	4.79E-2
19	**owl:unionOf**	3.15E-2
20	**owl:SymmetricProperty**	3.13E-2
21	**owl:TransitiveProperty**	2.98E-2
22	**owl:someValuesFrom**	2.13E-2
23	**rdf:_***	1.42E-2
24	**owl:allValuesFrom**	2.98E-3
25	**owl:minCardinality**	2.43E-3
26	**owl:maxCardinality**	2.14E-3
27	**owl:cardinality**	1.75E-3
28	**owl:oneOf**	4.13E-4
29	**owl:hasValue**	3.91E-4
30	**owl:intersectionOf**	3.37E-4
31	**owl:NamedIndividual**	3.37E-4

How much OWL is used in Linked Data? [LDOW'12]

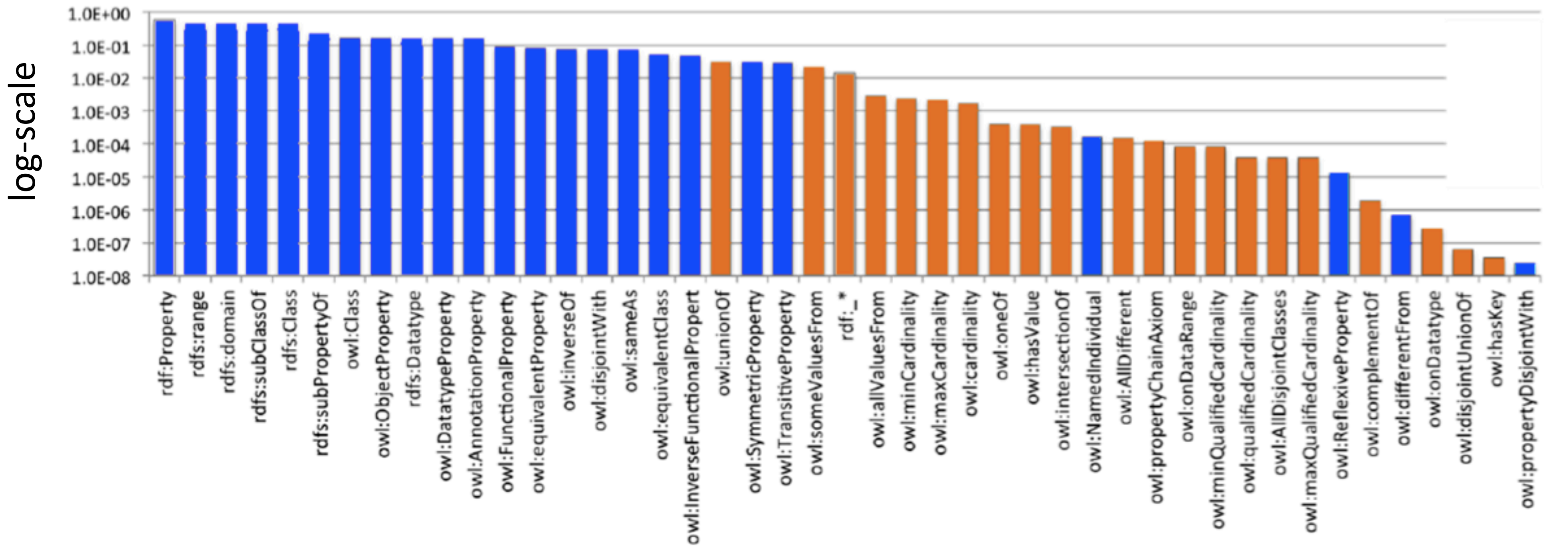
- RDF Schema features amongst the most prominently used
- OWL 2 features not yet used prominently

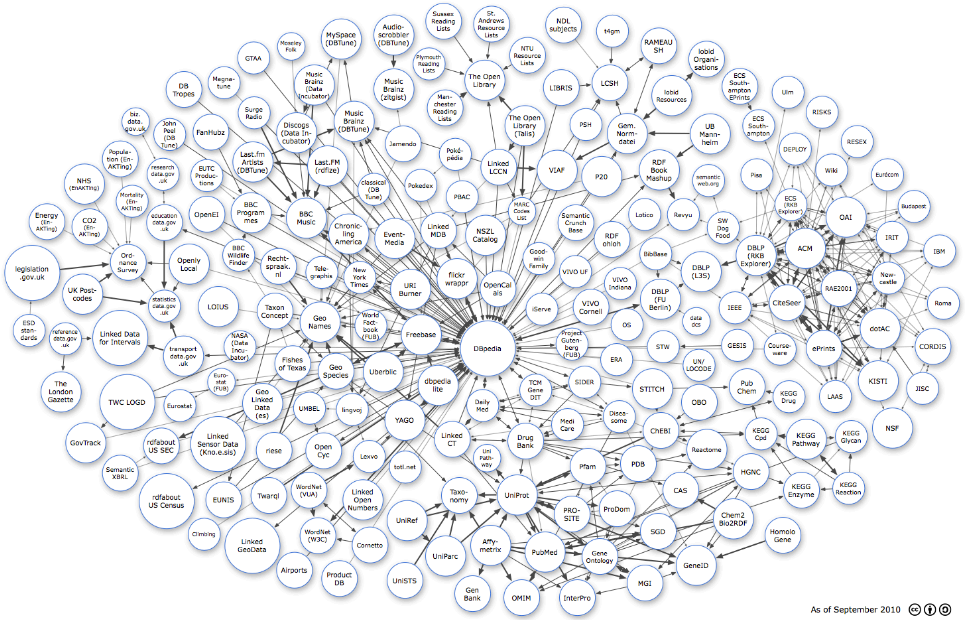
RDF | RDFS | OWL | OWL 2

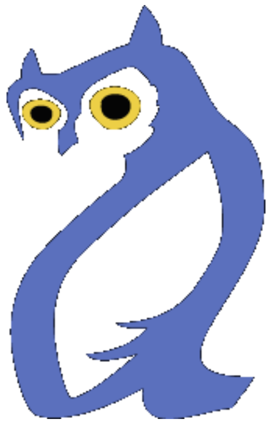


Main insight:

- RDF Schema features amongst the most prominently used
- OWL 2 features not yet used prominently
- Most *used* features can be implemented efficiently using **parallelizable, rule-based inference.** [IJSWIS'09]







Intuition: **MAYBE LINKED DATA ONLY NEEDS A LITTLE OWL…**

(…for now)

[Hendler, '98]

How much OWL is used in Linked Data? [LDOW'12]

- RDF Schema features amongst the most prominently used
- OWL 2 features not yet used prominently

In all fairness:

This is only a snapshot…

… some OWL2 features could be quite useful to enrich current Linked Data vocabularies, see also [RR'09]

We presented this 3 years ago at ESWC… still only little OWL2 adoption in LOD vocabs since then…

OWL2

ESWC2010

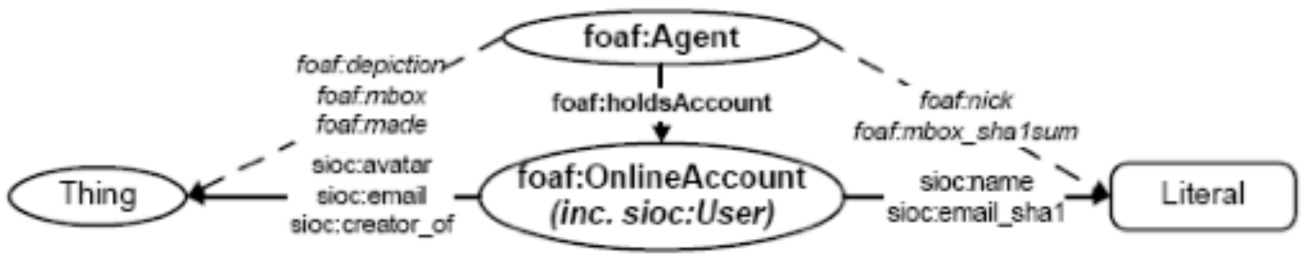
HOT! NEW!

Common ontologies on the Web don't use it a lot as of yet…

… but adds interesting functionality, potentially useful for Web ontologies, e.g.

- PropertyChains
  - E.g. could be useful to tie sioc:name and foaf:nick via foaf:holdsAccount:

```
foaf:nick owl:propertyChainAxiom (foaf:holdsAccount sioc:name)
```



Corollary2: … publishers still manage to make mistakes.

Examples…

Inconsistencies/wrong inferences on Web Data

4 main conjectures for possible reasons:

- – Publishers deliberately publish spoilt data ("SPAM")
- – Opinions differ
- – "URI-sense" ambiguities
- – **Accidently** wrong/inconsistent

Least common?

↓

Most common?

Publishers deliberately publish spoilt data ("SPAM")

- Examples:
  - a owl:differentFrom a .
  - foaf:knows rdfs:subPropertyOf owl:sameAs .
  - http://www.polleres.net/nasty.rdf
- Can occur for "testdata" being published, deliberate SPAM might become an issue, as the SW grows!

Opinions differ

- Fictitous Example Ontology:

Originofthings.example.org:

```
o1:surpremePower owl:disjointWith o1:naturalPhenomenom.
o1:originsFrom rdf:type owl:functionalProperty.
o1:god rdf:type o1:surpremePower.
o1:evolution rdf:type o1:naturalPhenomenom.
```

darwin.example.org:

```
ex:mankind o1:originsFrom o1:evolution .
```

creationism.example.org:

```
ex:mankind o1:originsFrom o1:god
```

FlyingSpaghettimonster.org

```
fsm:theSpaghettiMonster rdf:type surpremePower.
ex:mankind o1:originsFrom fsm:theSpaghettiMonster.
```

"URI-sense" ambiguities

```
<http://www.polleres.net>
    foaf:knows <http://apassant.net>
```

i.e., why do I have to use a different URI for myself and my homepage?

Many people don't understand/like this and make mistakes.

But is this really a mistake or a design error?

Accidentially inconsistent data

This appeared a while ago in our crawls:

```
Source1 (faulty):
TimBL foaf:homepage <http://www.w3.org>
TimBL rdf:type foaf:Person.
W3.org:
W3C foaf:homepage <http://www.w3.org>
W3C rdf:type foaf:Organisation.
```

Did occur in our Web crawls at some point, sometimes people don't have the right semantics in mind!

- Suspiciously resembles problems with e.g. flawed HTML … browsers, normal search engines still have to deal with it...

Accidently wrong (non-inconsistent data)

- FOAF Ontology:

```
foaf:mbox rdf:type owl:InverseFunctionalProperty .
```

- Careless FOAF exporters produce something like this for any empty email address:

```
ex:alice foaf:mbox "mailto:"
ex:bob foaf:mbox "mailto:"
…
```

Consequence: InverseFunctionalProperty reasoning on Web Data potentially equates too many things! Dangerous!

Not all of these were real examples, of course, for a more systematic analysis of Linked Data conformance, see [JWS'12a,LDOW'10]

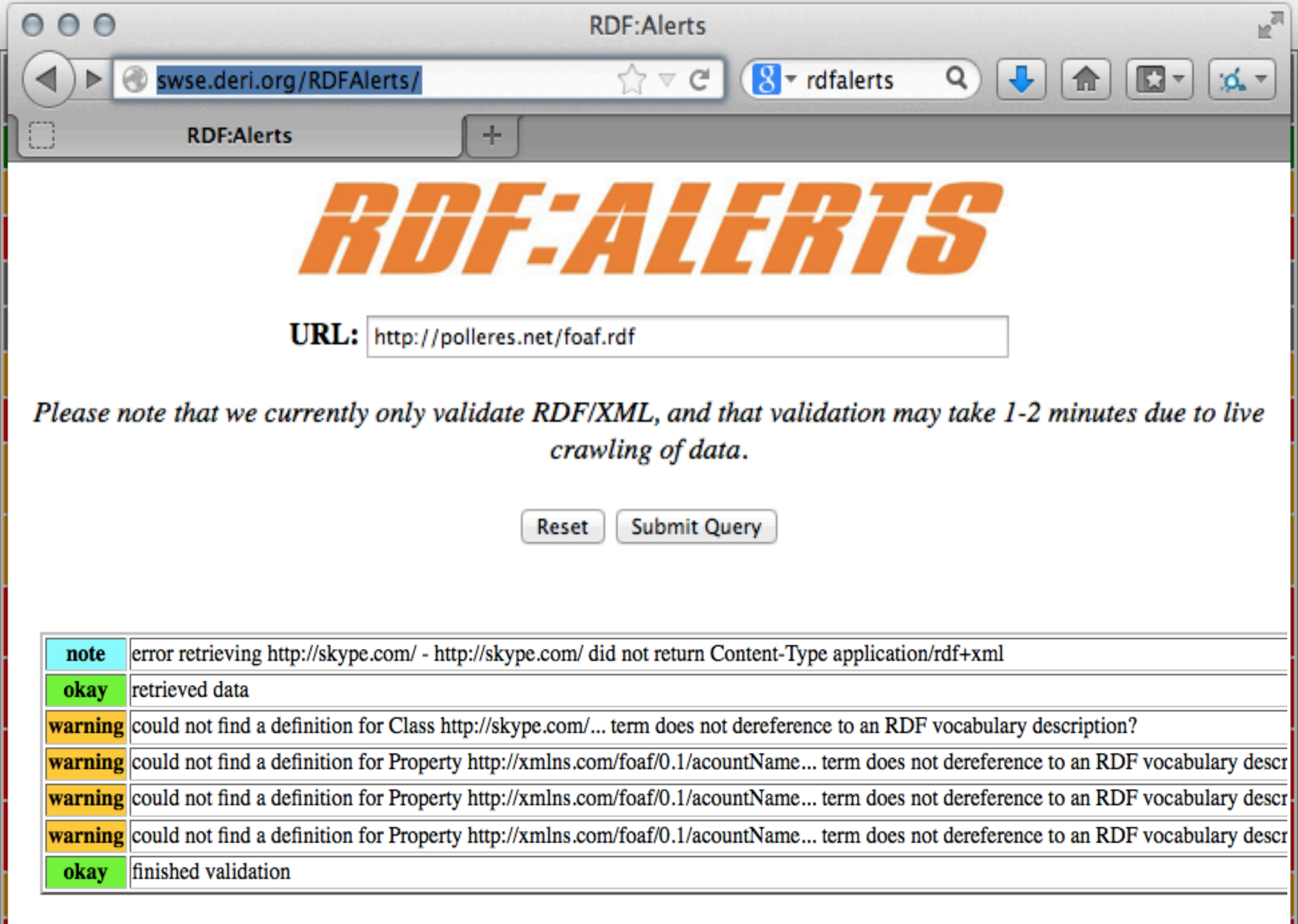
BTW, Didn't even mention ill-typed datatype literals here (very common…)

RDF:ALERTS

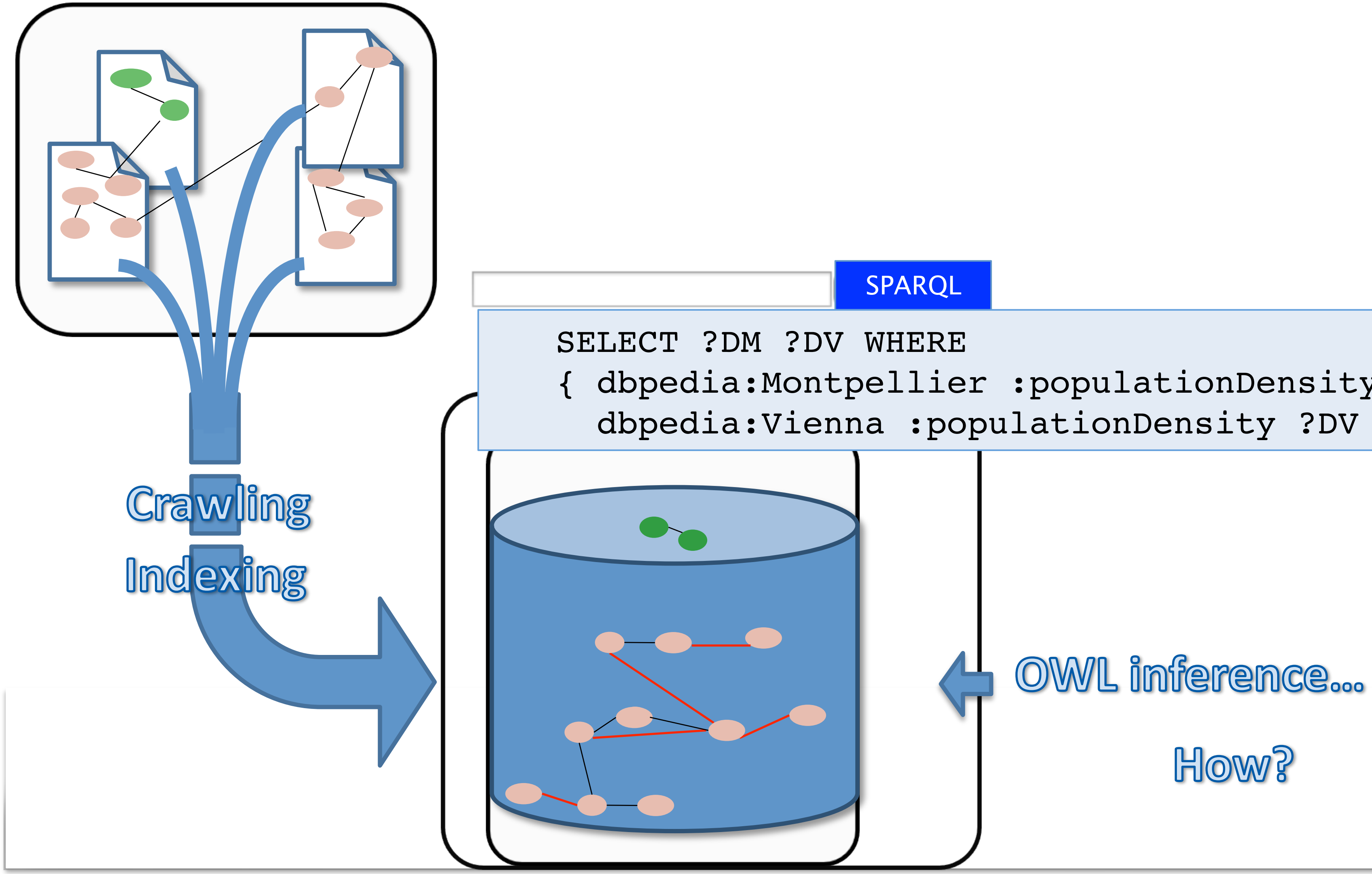
Your mission, should you decide to accept it, would be to make the Semantic Web clean …

---

http://swse.deri.org/RDFAlerts/

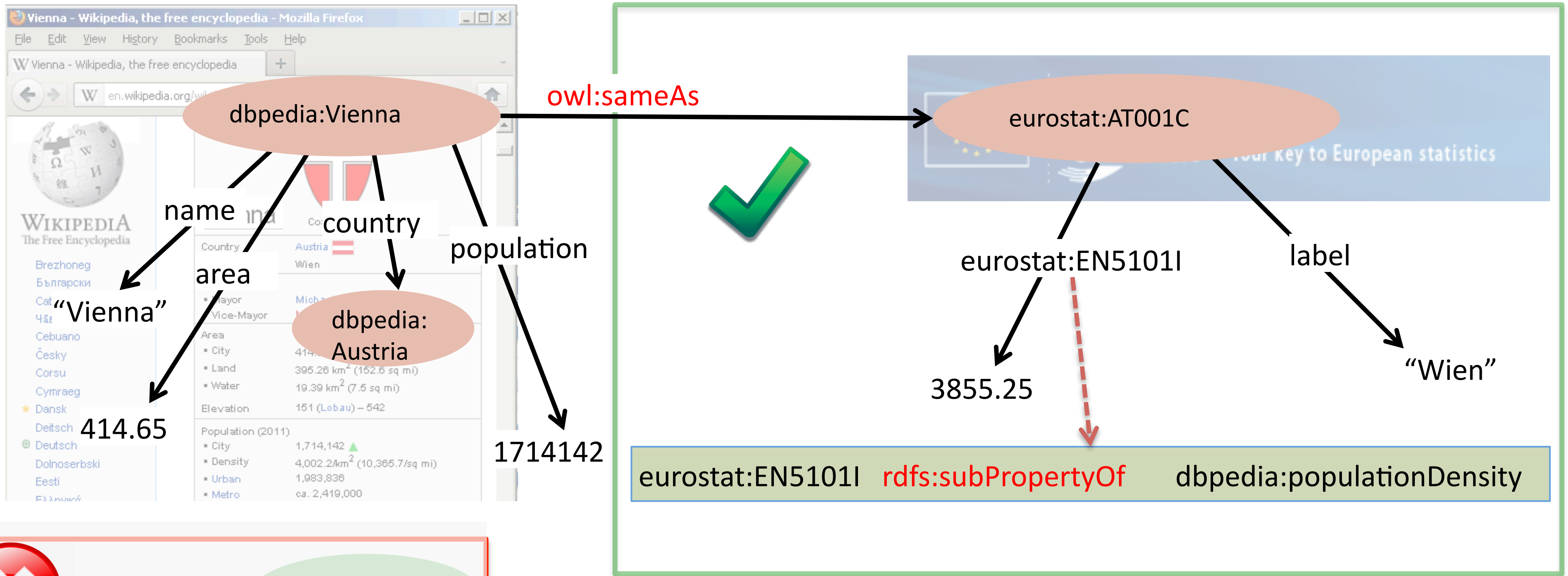


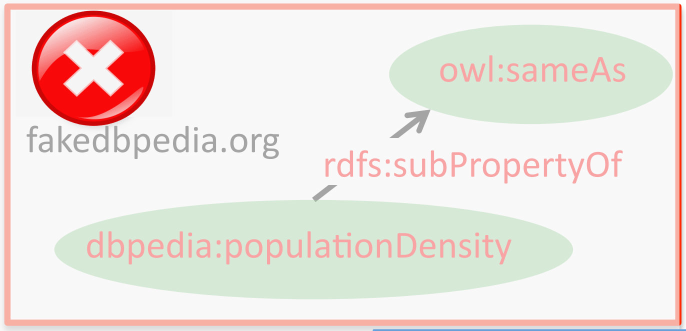
Conclusion: This won't scale to the Web…




```
SELECT ?DM ?DV WHERE
{ dbpedia:Montpellier :populationDensity ?DM .
  dbpedia:Vienna :populationDensity ?DV . }
```

How to possibly avoid nasty inferences?





Scalable Authoritative OWL Reasoning [IJSWIS'09] to the rescue!

… Goal:make inference **robust** & **scalable** to Web data.

```
SELECT ?DM ?DV
WHERE { dbpedia:Montpellier :populationDensity ?DM .
        dbpedia:Vienna :populationDensity ?DV .
      }
```

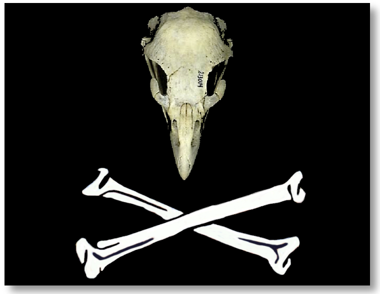
Scalabe Reasoning: Focus on Rules without ABox joins

DL Syntax	Rule
	$\mathcal{G}0$: NO A-BOX PATTERNS IN ANTECEDENT
$\{o_i....o_n\}$	?C :oneOf (?o_1 ... ?o_n) . ⇒ ?o_1 ... ?o_n a ?C .
	$\mathcal{G}1$: ONE A-BOX PATTERN IN ANTECEDENT
$C \sqsubseteq D$	?C rdfs:subClassOf ?D . ?s a ?C . ⇒ ?s a ?D .
$C \equiv D$	?C :equivalentClass ?D . ?s a ?C . ⇒ ?s a ?D .
	?C :equivalentClass ?D . ?s a ?D . ⇒ ?s a ?C .
$P \sqsubseteq Q$	?P rdfs:subPropertyOf ?Q . ?s ?P ?o . ⇒ ?s ?Q ?o .
$P \equiv Q$	?P :equivalentProperty ?Q . ?s ?P ?o . ⇒ ?s ?Q ?o .
	?P :equivalentProperty ?Q . ?s ?Q ?o . ⇒ ?s ?P ?o .
$P \equiv P_0^-$	?P :inverseOf ?Q . ?s ?P ?o . ⇒ ?o ?Q ?s .
	?P :inverseOf ?Q . ?s ?Q ?o . ⇒ ?o ?P ?s .
$\top \sqsubseteq \forall P^-.C$	?P rdfs:domain ?C . ?s ?P ?o . ⇒ ?s a ?C .
$\top \sqsubseteq \forall P.C$	?P rdfs:range ?C . ?s ?P ?o . ⇒ ?o a ?C .
$P \equiv P^-$	?P a :SymmetricProperty . ?s ?P ?o . ⇒ ?o ?P ?s .
$\exists P.x$	?C :hasValue ?x; :onProperty ?P . ?y ?P ?x . ⇒ ?y a ?C .
	?C :hasValue ?x; :onProperty ?P . ?y a ?C . ⇒ ?y ?P ?x .
$C_1 \sqcup ... \sqcup C_n$	?C :unionOf (?C_1...?C_i...?C_n) . ?x a ?C_i . ⇒ ?x a ?C .
$(\geq 1P)$	?C :minCardinality 1; :onProperty ?P . ?x ?P ?y . ⇒ ?x a ?C .
$C_1 \sqcap ... \sqcap C_n$	?C :intersectionOf (?C_1 ... ?C_n) . ?y a ?C . ⇒ ?y a ?C_1, ..., ?C_n .
$C_1 \sqcap ... \sqcap C_n$	?C :intersectionOf (?C_1) . ?y a ?C_1 . ⇒ ?y a ?C .

*From [IJSWIS'09] The rules applied including statements considered to be T-Box, elements which must be **authoritatively** spoken for (including for bnode OWL abstract syntax), and output count.*

Abox inferences only! Can be distributed! Several Optimizations in [ISWC2010]

Authoritative Reasoning



Ontology "Hijacking"

- Document **D** authoritative for concept **C** iff:
  - **C** not identified by a URI
    - OR
  - De-referenced URI of **C** coincides with or redirects to **D**
  - FOAF spec authoritative for `foaf:Person` ✓
  - MY Ont not authoritative for `foaf:Person` ✗
- Only allow extension in authoritative documents
  - `my:Person rdfs:subClassOf foaf:Person .` (MY Ont) ✓
- BUT: Reduce obscure memberships
  - `foaf:Person rdfs:subClassOf my:Person .` (MY Ont) ✗
- Similarly for other T-Box statements.
- In-memory T-Box stores authoritative values for rule execution

A Semantic Web Search Engine

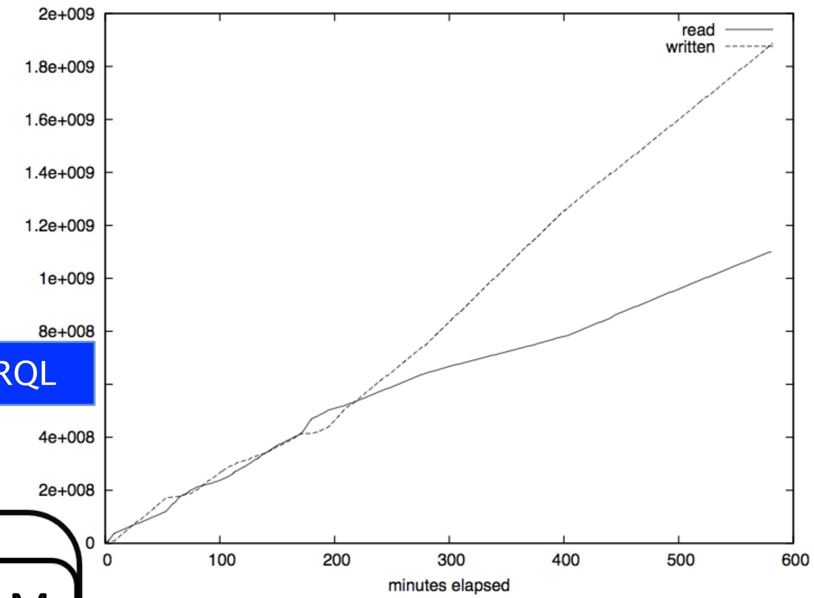
- … based on these insights we have implemented SWSE **[JWS'11]**

SWSE

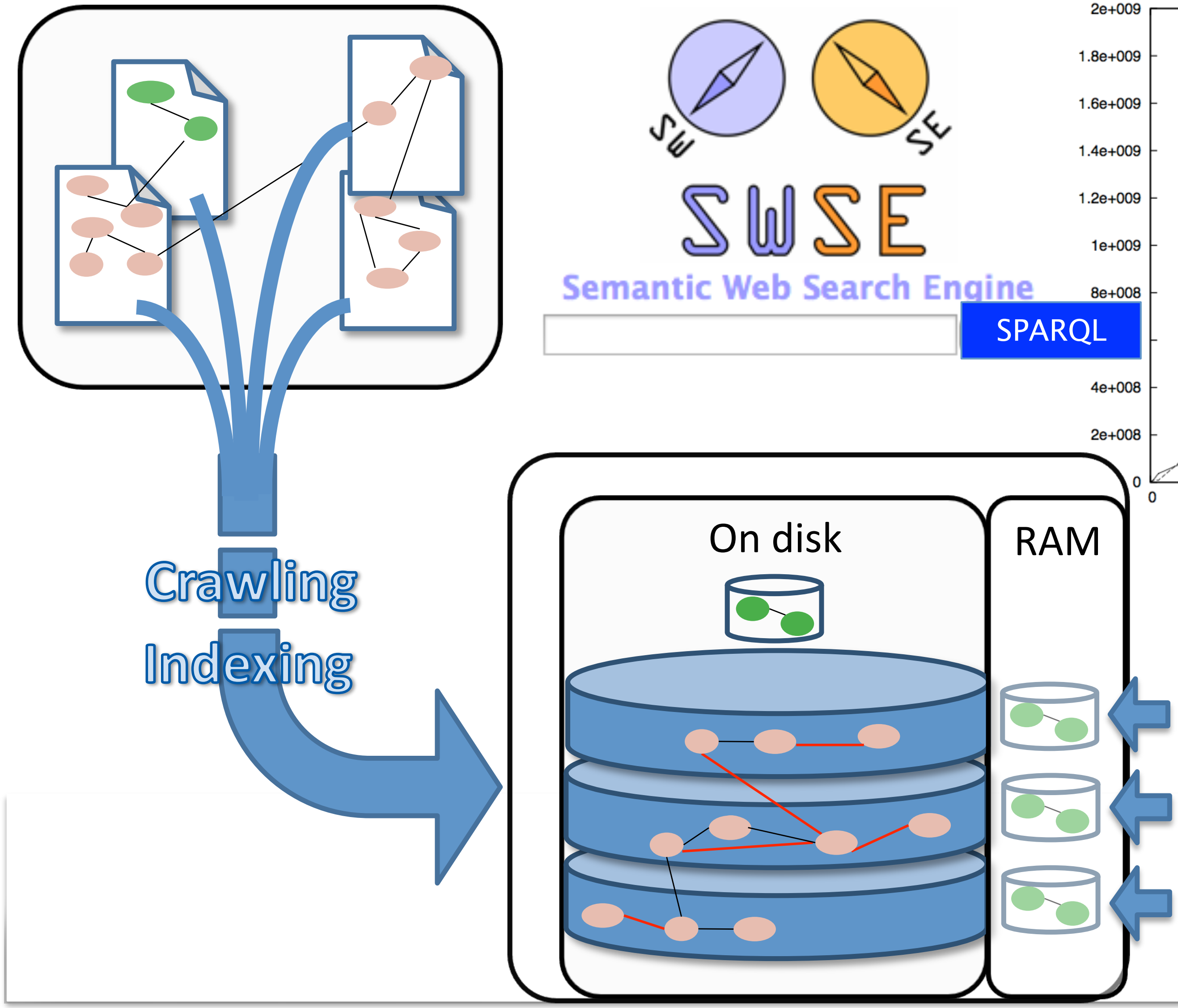
Semantic Web Search Engine

SPARQL

SAOR makes inference **robust** & **scalable** to Web data:



(on a single machine, goes down near linear in parallel mode)



Scalable
Authoritative
OWL Reasoner
(SAOR)

Are we done?

Similar in spirit to SAOR: "Quarantined Reasoning", cf. [RR'11]

indice THE SEMANTIC WEB INDEX

"Which city of Montpellier and Vienna has the higher population density?" ✓

SWSE

indice THE SEMANTIC WEB INDEX

"Latest news on NYT about technology companies with a revenue greater than 10B EUR" ✗

```
SELECT * WHERE
{ ?C rdf:type NYT:Org .
  ?C dbpedia:revenue ?R .
  ?C NYT:latestArticle ?A .
  FILTER( ?R > 10000000000 )  }
```

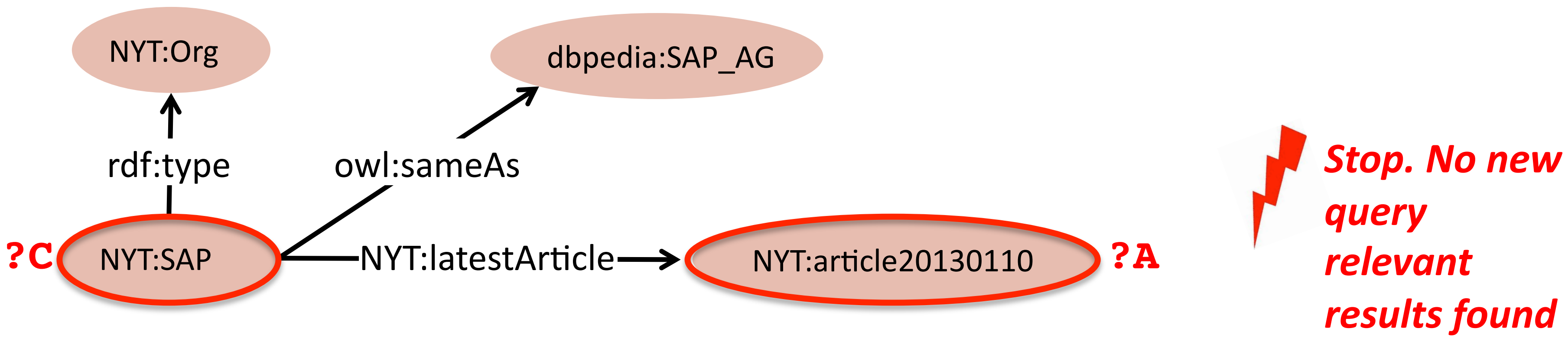
Can't keep up with changes!

Idea: “Live”-Linked Data Querying

- Basic idea (originally proposed by Hartig & Bizer)
  - Start with URIs in a SPARQL query
  - Interleave crawl + query processing

```
SELECT * WHERE
{ ?C rdf:type NYT:Org .
  ?C dbpedia:revenue ?R .
  ?C NYT:latestArticle ?A.
FILTER (?R > €10000000000 )  }
```

HTTP GET NYT:Org → HTTP GET NYT:SAP → HTTP GET NYT:article20130128_1

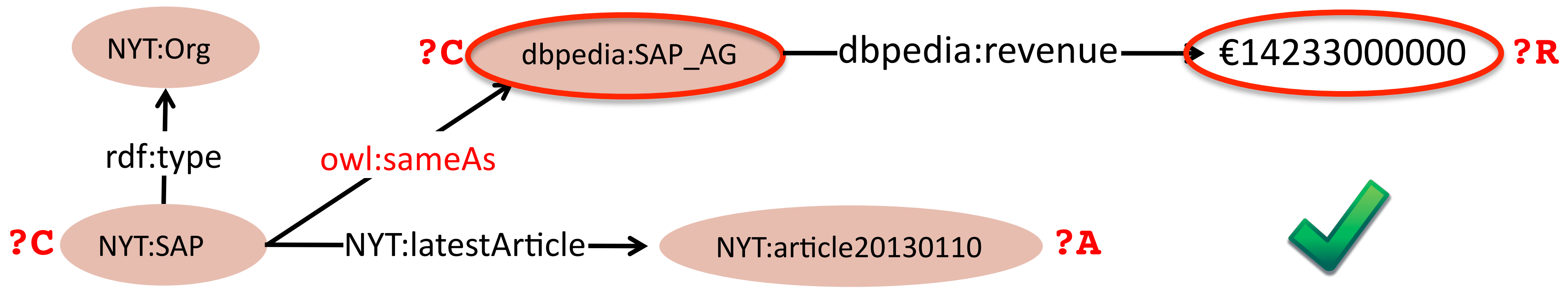


Improved "Live"-Linked Date Querying

- Our approach [RR'12]
- Basic idea:
  - Start with URIs in a SPARQL query
  - **Interleave crawl + query processing**
    + Lightweight OWL Reasoning

```
SELECT * WHERE
{ ?C rdf:type NYT:Org .
  ?C dbpedia:revenue ?R .
  ?C NYT:latestArticle ?A.
FILTER (?R > €10000000000)  }
```

HTTP GET NYT:Org → HTTP GET NYT:SAP → HTTP GET dbpedia:SAP

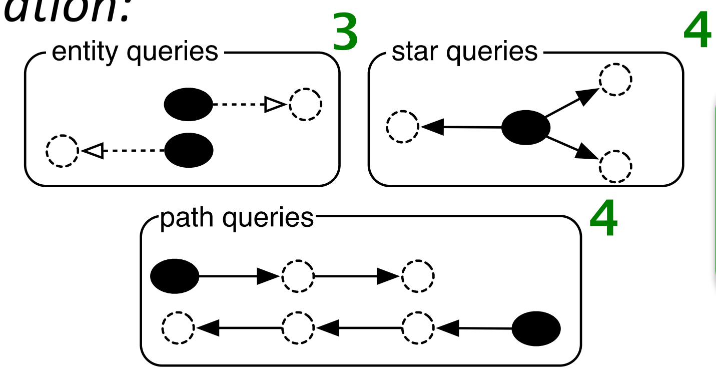


Again: Does this work "in the wild"?

1) *Benchmark generation:*

BTC 2011

QWalk:
Random walk based query generation.



1100 queries
100 each for 11 "*typical*" shapes

2) *Results:*

- **RDFS** reasoning: *many queries with average impact*
  - In 8/11 query classes more than 50% avg. result increase
  - in 4/11 query classes a time increase of 50%
- **OWL** sameAs: *view queries with high impact*
  - In 2/11 query classes more than 50% avg. result increase
  - In 2/11 query classes a time increase of 50%

Overall our reasoning extensions improved in 8 of 11 query classes the average result/time ratio (throughput)

Are we done?

"Which city of Montpellier and Vienna has the higher population density?"

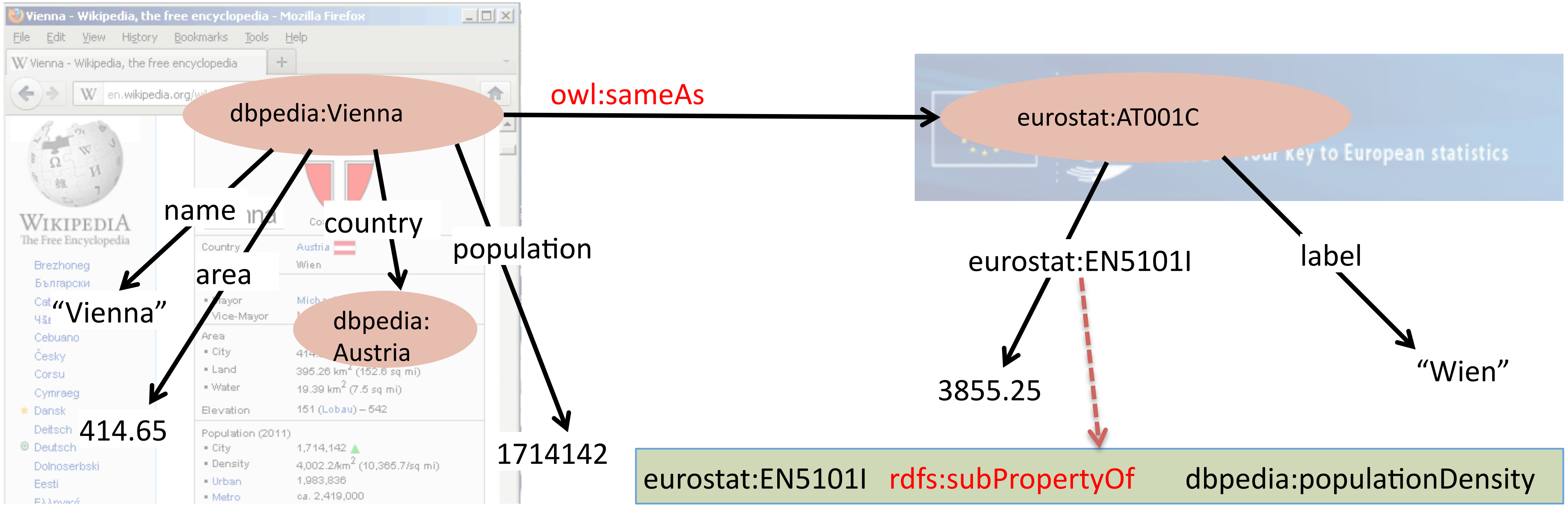
"Latest news on NYT about technology companies with a revenue greater than 10B EUR"

Depends on the point of view…

… e.g.: What is the sweetspot between **centralised & decentralised/live** Linked Data querying approaches?

Plus…

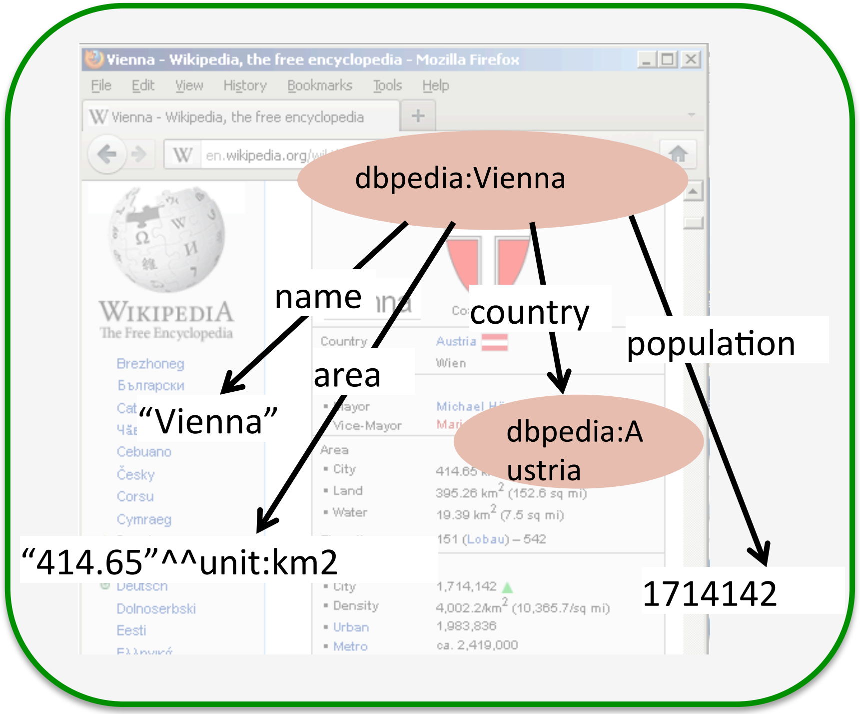
… Actually, we were cheating:



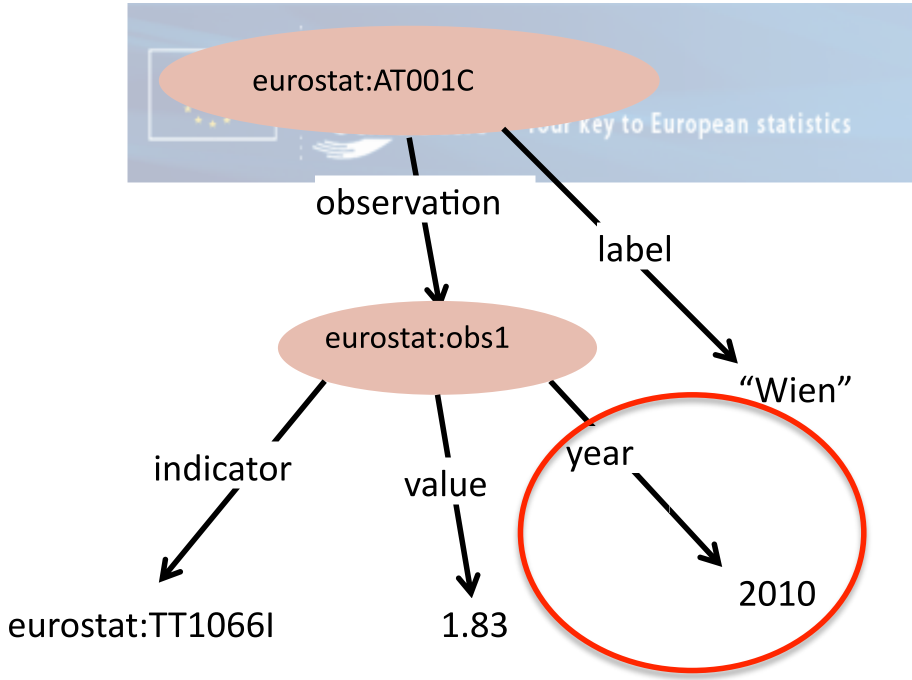
"Which city of Montpellier and Vienna has the higher population density?"

```
SELECT ?DM ?DV
WHERE { dbpedia:Montpellier :populationDensity ?DM .
        dbpedia:Vienna :populationDensity ?DV .
      }
```

Example: Actually, we were cheating…



Data looks rather like this…



"Which city of Montpellier and Vienna has the higher population density?"

Can't we still answer this query? Just using the Dbpedia data?

We know: **dbpedia:opulationDensity = dbpedia:population / dbpedia:area**

Another example is unit conversion from this morning:

elephantWeight = elphantWeightImperial * 2.204

or priceUSD = priceEUR * 1.2850

We call these "**propery equation**" … stay tuned for the main track [ESWC'13]

Take-home messages:

Lots of interesting real-world Datasets out there as Linked Data!

- can benefit from OWL and Reasoning
- (even more Open Data in other formats, BTW)

Structured Query anwsering over Linked Data needs:

- **Robust & Scalable Reasoning** (à la SAOR)
- Both **centralised & decentralised** approaches
- Maybe more than OWL (e.g. something like property equations)?

Vienna

Population density 2012
4002.2 p/km2 (Source: http://dbpedia.org/)
Population density 2004
3855.25 p/km2
(Source: http://www.urbanaudit.org/)
Population density 2001
3735.24 p/km2
(Source: http://www.urbanaudit.org/)
Population density 1999
3735.2 p/km2 (Source: http://eurostat.linked-statistics.org/)

References:

[RR'12] Jürgen Umbrich, Aidan Hogan, Axel Polleres, Stefan Decker: Improving the Recall of Live Linked Data Querying through Reasoning. *In 6th International Conference on WebReasoning and Rule Systems* (RR2012)

[LDOW'12] Birte Glimm, Aidan Hogan, Markus Krötzsch, Axel Polleres: OWL: Yet to arrive on the Web of Data? *In Linked Data on the Web* (LDOW2012)

[JWS'11] Aidan Hogan, Andreas Harth, Jürgen Umbrich, Sheila Kinsella, Axel Polleres, Stefan Decker: Searching and browsing Linked Data with SWSE: The Semantic Web Search Engine. *JWS*. 9(4): 365-401 (2011)

[LDOW'10] Aidan Hogan, Andreas Harth, Alexandre Passant, Stefan Decker, Axel Polleres: Weaving the Pedantic Web. LDOW 2010.

[JWS'12a] Aidan Hogan, Jürgen Umbrich, Andreas Harth, Richard Cyganiak, Axel Polleres, Stefan Decker: An empirical survey of Linked Data conformance. J. Web Sem. 14: 14-44 (2012)

[ESWC'13] Stefan Bischof, Axel Polleres: RDFS with Attribute Equations via SPARQL Rewriting. ESWC 2013: 335-350

[RR'11] Renaud Delbru, Giovanni Tummarello, Axel Polleres: Context-Dependent OWL Reasoning in Sindice - Experiences and Lessons Learnt. RR 2011: 46-60

[ISWC'10] Aidan Hogan, Jeff Z. Pan, Axel Polleres, Stefan Decker: SAOR: Template Rule Optimisations for Distributed Reasoning over 1 Billion Linked Data Triples. ISWC (1) 2010: 337-353

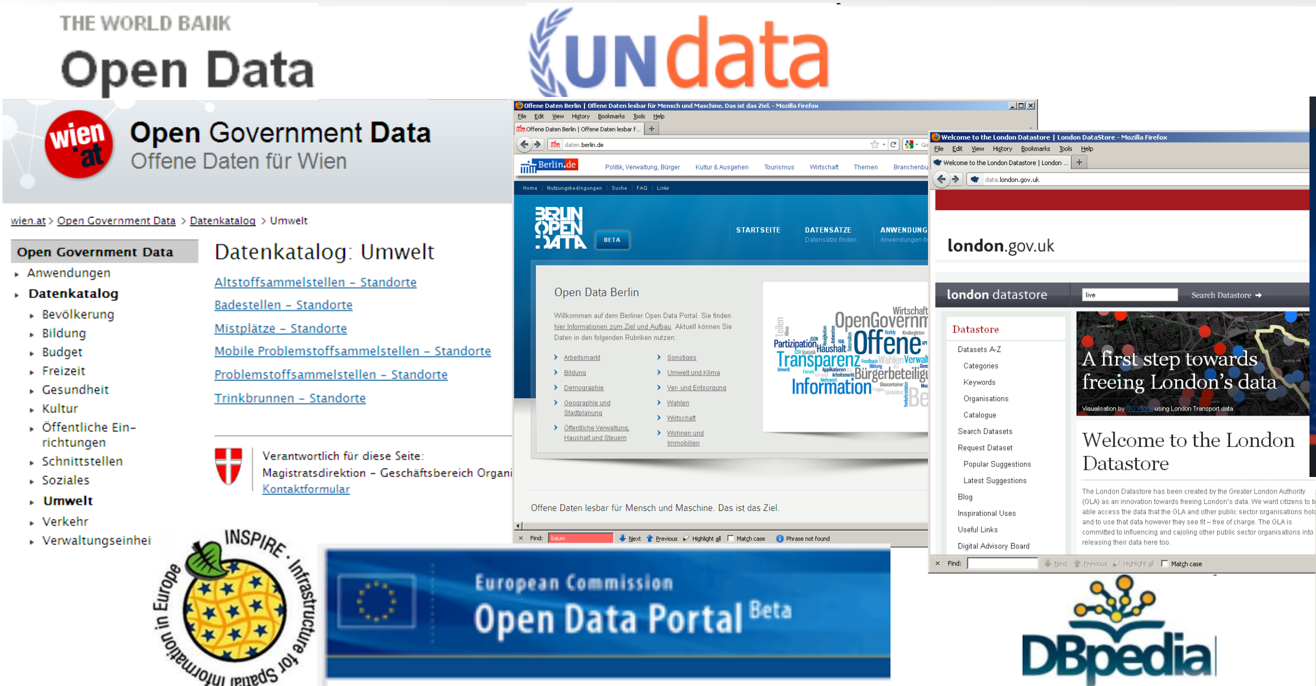
[RR'09] Aidan Hogan, Stefan Decker: On the Ostensibly Silent 'W' in OWL 2 RL. RR 2009: 118-134

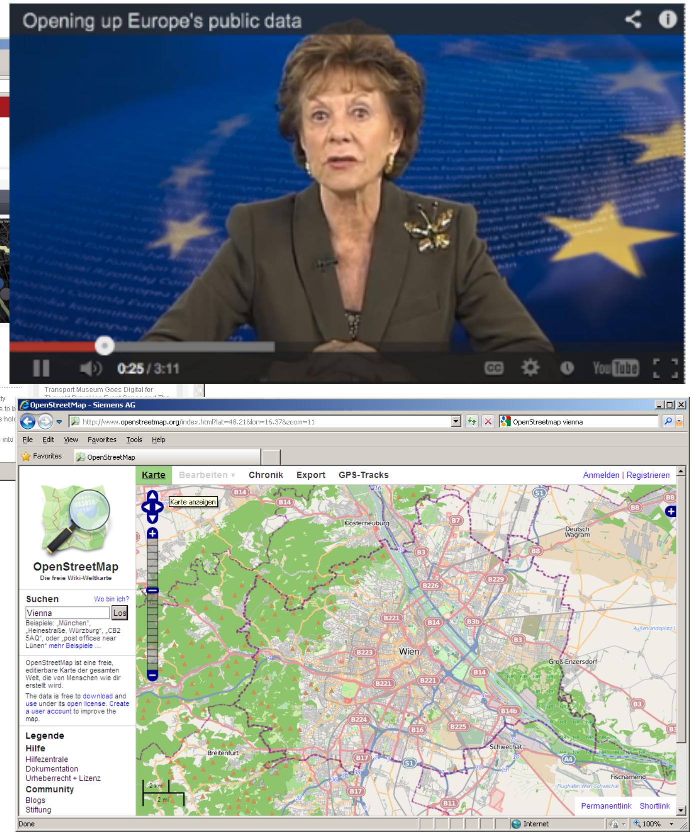
[IJSWIS'09] Aidan Hogan, Andreas Harth, Axel Polleres: Scalable Authoritative OWL Reasoning for the Web. *Int. Journal On Semantic Web & Information Systems* 5(2): 49-90 (2009)

…(more on www.polleres.net/publications.html)

Backup-slides

Open Data
Trends & Future





DIRECTIVE 2007/2/EC INSPIRE

- Open Data: Typically very liberal licenses (variants of CC)
- Many formats, varying quality, harmonization starting
- Mostly by online communities or public bodies (cities, communities, governments, UN,…)
  - Currently focused mostly in SMEs to take advantage of that data
- vs. Publicly available data: e.g. NYT is public but not free/not license free
- vs. Enterprise (Linked) Data

Linked Data Ontologies = RDF Vocabularies (OWL, RDFS)

GeoNames

Google code labs

schema.org

DOAP

DBpedia

FOAF

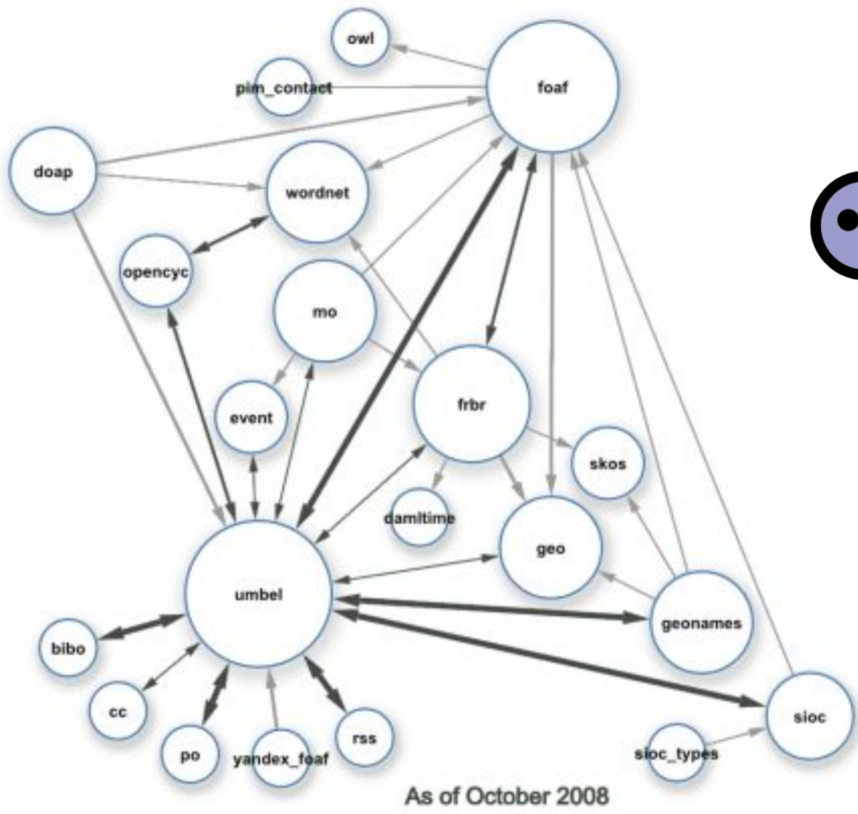
DublinCore

GoodRelations
The Web Ontology for E-Commerce

music ontology

SKOS

umbel



...

Image from http://blog.dbtune.org/public/.081005_lod_constellation_m.jpg:; Giasson, Bergman

Side remark: *Only a small fraction of Linked data is OWL/RDFS ontologies (less than ~0.1% [IJSWIS'2009, Hogan 2010])*

OWL Reasoning: Complexity

- http://www.w3.org/TR/2009/REC-owl2-profiles-20091027/

Language	Reasoning Problems	Taxonomic Complexity	Data Complexity	Query Complexity	Combined Complexity
OWL 2 RDF-Based Semantics	Ontology Consistency, Class Expression Satisfiability, Class Expression Subsumption, Instance Checking, Conjunctive Query Answering	Undecidable	Undecidable	Undecidable	Undecidable
OWL 2 Direct Semantics	Ontology Consistency, Class Expression Satisfiability, Class Expression Subsumption, Instance Checking	2NEXPTIME-complete (NEXPTIME if property hierarchies are bounded)	Decidable, but complexity open (NP-Hard)	Not Applicable	2NEXPTIME-complete (NEXPTIME if property hierarchies are bounded)
	Conjunctive Query Answering	Decidability open	Decidability open	Decidability open	Decidability open

And even for OWL RL PTIME-complete…. i.e., often too expensive.

Related Works:

- Scalable OWL Reasoning
  - Jesse Weaver, James A. Hendler: Parallel Materialization of the Finite RDFS Closure for Hundreds of Millions of Triples. International Semantic Web Conference 2009: 682-697
  - Vladimir Kolovski, Zhe Wu, George Eadon: Optimizing Enterprise-Scale OWL 2 RL Reasoning in a Relational Database System. International Semantic Web Conference (1) 2010: 436-452
  - Jacopo Urbani, Spyros Kotoulas, Jason Maassen, Frank van Harmelen, Henri E. Bal: WebPIE: A Web-scale Parallel Inference Engine using MapReduce. J. Web Sem. 10: 59-75 (2012)

RDFS only, syntactic data. Nice hardware.

Similar ideas to ours in terms of rule optimization by ORACLE (parallel to our results)

Similar parallelization techniques to ours, 100b triples, but confined data with small ontologies. No Web Data. Nice: exchange of results for joins in Map-reduce jobs.

- Live Linked Data Querying
  - Olaf Hartig, Christian Bizer, Johann Christoph Freytag: Executing SPARQL Queries over the Web of Linked Data. International Semantic Web Conference 2009: 293-309
  - Günter Ladwig, Thanh Tran: Linked Data Query Processing Strategies International Semantic Web Conference (1) 2010: 453-469

No reasoning considered. 4 fixed queries, result throughput not considereed

Confined Data, no reasoning considered. 8 fixed queries, needs a source index