

Pulling Together, or How I Learned to Love the Semantic Web

Kate Byrne, School of Informatics, University of Edinburgh

14th November 2008

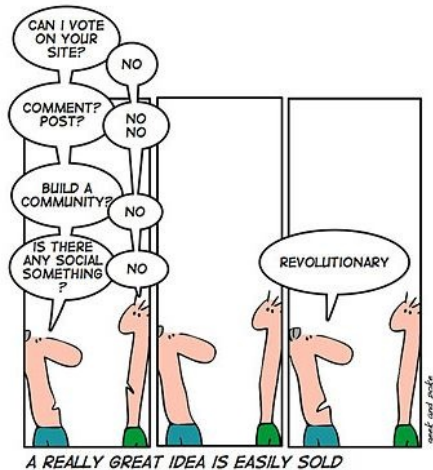
What is the Semantic Web?

- Aka *Web of Data*, *Giant Global Graph*, *Web 3.0*
- The “Document Web” connects HTML documents . . .
- . . . the Semantic Web connects RDF data nodes



Tim Berners Lee on the Semantic Web

Web 3.0 vs Web 2.0



What Does the Semantic Web Do?

- The Semantic Web is for machines to read
- Gather information and reason over it using rules
 - travel bookings
 - medical diagnosis
 - “ambient intelligence”
 - cultural tourism
- Underlying framework with numerous applications
 - many of them benign ☺

What Does the Semantic Web Do?

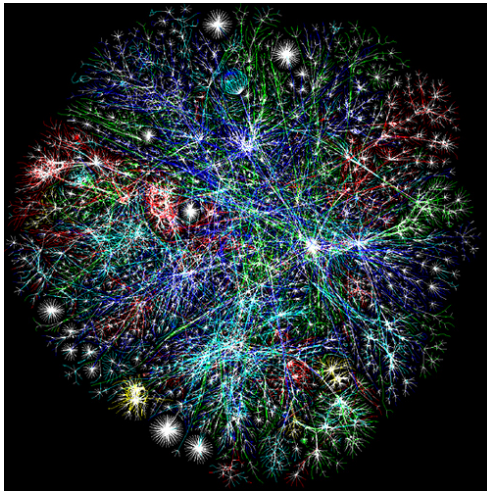
- The Semantic Web is for machines to read
- Gather information and reason over it using rules
 - travel bookings
 - medical diagnosis
 - “ambient intelligence”
 - cultural tourism
- Underlying framework with numerous applications
 - many of them benign ☺

What Does the Semantic Web Do?

- The Semantic Web is for machines to read
- Gather information and reason over it using rules
 - travel bookings
 - medical diagnosis
 - “ambient intelligence”
 - cultural tourism
- Underlying framework with numerous applications
 - many of them benign 😊

The Giant Global Graph

(<http://www.opte.org/maps/tests/>)



Sun Nov 23 22:03:42 PST 2003

It's Triplets All The Way Down

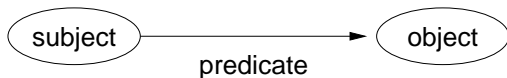


RDF – Resource Description Framework

- “Facts” expressed as subject–property–object triples:
- Resources: nodes or arcs (edges) with URIs
- Resource nodes can be subject – so can link triples together
- Literals (strings, numbers) can only be object

RDF – Resource Description Framework

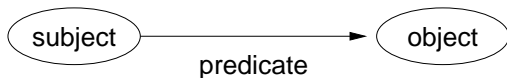
- “Facts” expressed as subject–property–object triples:



- Resources: nodes or arcs (edges) with URIs
- Resource nodes can be subject – so can link triples together
- Literals (strings, numbers) can only be object

RDF – Resource Description Framework

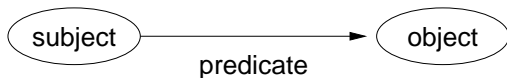
- “Facts” expressed as subject–property–object triples:



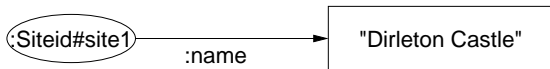
- Resources: nodes or arcs (edges) with URIs
- Resource nodes can be subject – so can link triples together
- Literals (strings, numbers) can only be object

RDF – Resource Description Framework

- “Facts” expressed as subject–property–object triples:



- Resources: nodes or arcs (edges) with URIs
- Resource nodes can be subject – so can link triples together
- Literals (strings, numbers) can only be object



@prefix : <<http://www.ltg.ed.ac.uk/tether/>> .

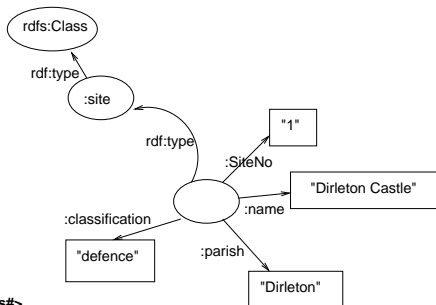
Relational Database to RDF Graph

RDB2RDF by “Table as Class; Column as Predicate” method

SITE

siteNo	name	parish	classification
1	<i>Dirleton Castle</i>	<i>Dirleton</i>	<i>defence</i>
2	Dirleton Cottage	Dirleton	residential
3	Drem Airfield	Dirleton	military
4	Jamie's Neuk	Dirleton	military

@prefix : <http://www.ltg.ed.ac.uk/tether/> .
 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .



RDB2RDF Conversion

Converting relational data to RDF is straightforward.

RDB2RDF Conversion

Converting relational data to RDF is ^{not} straightforward.

Some of the Pitfalls in RDB2RDF

1. Literals (strings) can't be subject nodes – so use URIs – but:

- Photo description – '#5: 6"x4" neg B&W'

<http://www.ex.com/Pdesc#%235:%206%22x4%22%20neg%2C%20B%26W>

2. Take care with URI generation:

- is <http://www.example.com/place/edinburgh> the same resource as <http://www.example.com/city/edinburgh?>

3. Beware redundant RDF nodes where relational tables join

- 3 *million* redundant triples for my dataset

Some of the Pitfalls in RDB2RDF

1. Literals (strings) can't be subject nodes – so use URIs – but:
 - Photo description – '#5: 6"x4" neg, B&W'
<http://www.ex.com/Pdesc#%235:%206%22x4%22%20neg%2C%20B%26W>
2. Take care with URI generation:
 - is <http://www.example.com/place/edinburgh> the same resource as <http://www.example.com/city/edinburgh?>
3. Beware redundant RDF nodes where relational tables join
 - 3 *million* redundant triples for my dataset

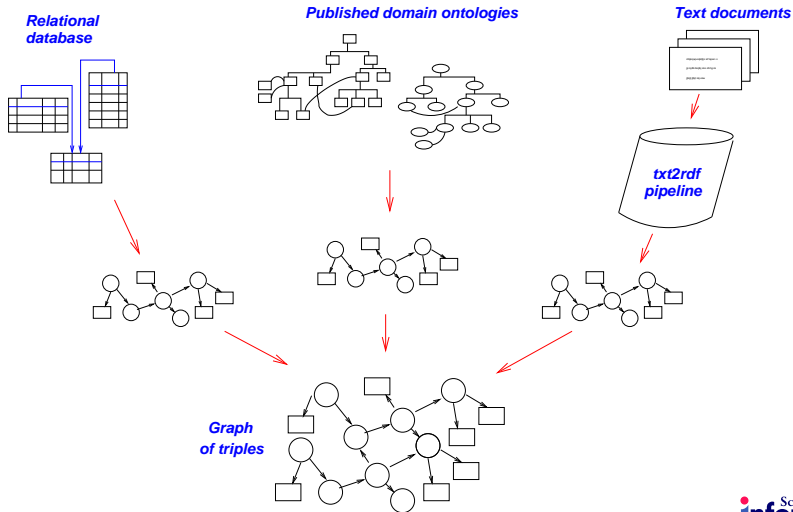
Some of the Pitfalls in RDB2RDF

1. Literals (strings) can't be subject nodes – so use URIs – but:
 - Photo description – '#5: 6"x4" neg, B&W'
<http://www.ex.com/Pdesc#%235:%206%22x4%22%20neg%2C%20B%26W>
2. Take care with URI generation:
 - is <http://www.example.com/place/edinburgh> the same resource as <http://www.example.com/city/edinburgh>?
3. Beware redundant RDF nodes where relational tables join
 - 3 *million* redundant triples for my dataset

Some of the Pitfalls in RDB2RDF

1. Literals (strings) can't be subject nodes – so use URIs – but:
 - Photo description – '#5: 6"x4" neg, B&W'
<http://www.ex.com/Pdesc#%235:%206%22x4%22%20neg%2C%20B%26W>
2. Take care with URI generation:
 - is <http://www.example.com/place/edinburgh> the same resource as <http://www.example.com/city/edinburgh>?
3. Beware redundant RDF nodes where relational tables join
 - 3 *million* redundant triples for my dataset

My Own Work – *Tether*



Data Collection from RCAHMS

The Royal Commission on the Ancient and Historical Monuments of Scotland

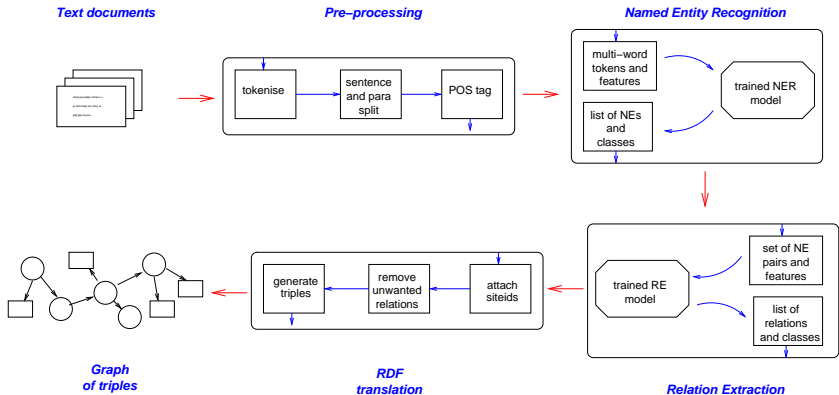
- Founded in February 1908
- <http://www.rcahms.gov.uk/>
- One of Scotland's 6 National Collections
- The “memory keeper” for Scotland

- Mission –
 - **survey** the built environment
 - **maintain a record** of buildings and archaeological sites
 - **promote understanding** of the material



Argyle St 40 years ago: <http://www.rcahms.../canmore...arcnumlink=669215>

NLP – Text to RDF “Pipeline”



Named Entity Recognition – Machine Learning Method

1. Decide the Named Entity classes you want to find
 - people, places, organisations, etc.
2. Annotate a set of training documents with NEs
 - eg mark “Glasgow” as *PLACE*
3. Train a **classifier** to learn the important **features**
 - the classifier builds a model of what NEs are like
4. Test the model against new documents

Named Entity Recognition – Machine Learning Method

1. Decide the Named Entity classes you want to find
 - people, places, organisations, etc.
2. Annotate a set of training documents with NEs
 - eg mark “Glasgow” as *PLACE*
3. Train a **classifier** to learn the important **features**
 - the classifier builds a model of what NEs are like
4. Test the model against new documents

Named Entity Recognition – Machine Learning Method

1. Decide the Named Entity classes you want to find
 - people, places, organisations, etc.
2. Annotate a set of training documents with NEs
 - eg mark “Glasgow” as *PLACE*
3. Train a **classifier** to learn the important **features**
 - the classifier builds a model of what NEs are like
4. Test the model against new documents

Named Entity Recognition – Machine Learning Method

1. Decide the Named Entity classes you want to find
 - people, places, organisations, etc.
2. Annotate a set of training documents with NEs
 - eg mark “Glasgow” as *PLACE*
3. Train a **classifier** to learn the important **features**
 - the classifier builds a model of what NEs are like
4. Test the model against new documents

RCAHMS Text with Named Entities Marked

SHAPINSAY , HELLIAR HOLM

HY41NE 2 4843 1534 .

In the SE part of **the island** is a large **cairn** , which is certainly **prehistoric** , although the name is printed in ordinary type on the OS map . It is mainly composed of fairly large stones , and measures **66' by 60'** , with a height of about **8'** . On its S side are three large stones set on end at irregular intervals , which have apparently formed part of a chamber , long since destroyed . The landmark on the top is **modern** . **RCAHMS 1946** , visited **1928**

Not visited . **A S Henshall 1963**

HY 4843 1534 : A **chambered cairn** measuring about **18.0m** in diameter and **2.2m** high , surmounted by a **modern marker cairn** . The tops of seven of four pairs of slabs of an apparently stalled chamber are visible in a central depression . Oriented NW-SE , the passage width between the slabs is **0.4m** and the pairs of slabs are **1.7m** apart . No back slab is evident , but the plan is reminiscent of the **Orkney-Cromarty stalled cairn** at the **Hill of Shebster** (ND06SW 5) . Resurveyed at 1:2500 . Visited by OS (**AA**) **1 October 1972**

A **stalled cairn** , as described by field surveyor (**AA**) in **1972** . There are indications of the top of drystone walling on the S side of the eastern - most compartment . (Confirmed by **A S Henshall**) . Visited by OS (**JLD**) **18 May 1981**

Key

Place **Site** **Reference** **Size** **Link** **Person** **Timex** **Organisation** **Period** **Unknown**

Relation Extraction

- Look for relationships between Named Entities
 - *site* – *hasLocation* – *place*
 - “Helliær Holm” – *hasLocation* – Shapinsay
- Method similar to NE step:
 1. Decide what types of relation to look for
 2. Mark relations in training documents
 3. Train a classifier to spot key features of relations
 4. Test on new documents

RCAHMS Text with Relations Marked

[SOUTH WALLS] , [MISBISTER] , [THE LOFTS]

[ND38NW 29 centred 3325 8885]

Sites [recorded] during an [archaeological survey] undertaken on the lands of [the Loft] , [Longhope] , as part of the pilot scheme for the [Historic [Scotland]] [Farm] [Ancient] [Monument] Survey Grant Scheme . [ND 3311 8890] Two [small cairns] . [ND 3336 8889] [Cairn] . [ND 3339 8885] [Cairn] . [ND 3339 8886] [Clearance cairn] . [ND 3342 8884] [Sub-rectangular cairn] . [ND 3339 8883] [Well] Sponsors : [Historic [Scotland]] , [M J Jones] . [N Card] [1998]

Once We Have Relations...

- Examples of relations:
 - “The Lofts” – *hasLocation* – Misbister
 - “The Lofts” – *hasEvent* – recording
 - recording – *hasLocation* – “ND 3342 8884”
 - recording – *hasPatient* – “Sub-rectangular cairn”
- Looks familiar? *subject* – *property* – *object* triples
- Issues to deal with:
 - fitting to a coherent RDF schema
 - generating suitable URIs
 - linking to database records
 - normalising text strings
 - weeding out useless relations
- End product? RDF graph

Once We Have Relations...

- Examples of relations:
 - “The Lofts” – *hasLocation* – Misbister
 - “The Lofts” – *hasEvent* – recording
 - recording – *hasLocation* – “ND 3342 8884”
 - recording – *hasPatient* – “Sub-rectangular cairn”
- Looks familiar? *subject* – *property* – *object* triples
- Issues to deal with:
 - fitting to a coherent RDF schema
 - generating suitable URIs
 - linking to database records
 - normalising text strings
 - weeding out useless relations
- End product? RDF graph

Once We Have Relations...

- Examples of relations:
 - “The Lofts” – *hasLocation* – Misbister
 - “The Lofts” – *hasEvent* – recording
 - recording – *hasLocation* – “ND 3342 8884”
 - recording – *hasPatient* – “Sub-rectangular cairn”
- Looks familiar? *subject* – *property* – *object* triples
- Issues to deal with:
 - fitting to a coherent RDF schema
 - generating suitable URIs
 - linking to database records
 - normalising text strings
 - weeding out useless relations
- End product? RDF graph

Why Convert? – Interoperability

- Related information:
 - NAS & GRO: births, deaths, marriages – *Scotland's People*
 - RCAHMS: sites from Neolithic to Now – *Scotland's Places*
 - NMS: excavation finds, cultural objects
 - NLS: bibliographic material supporting all of it
- Interconnecting relational databases is hard:
 - you need to know the schema in detail
 - security issues
 - complex networking protocols – not http
 - whereas RDF was *designed* for data linking

Why Convert? – Interoperability

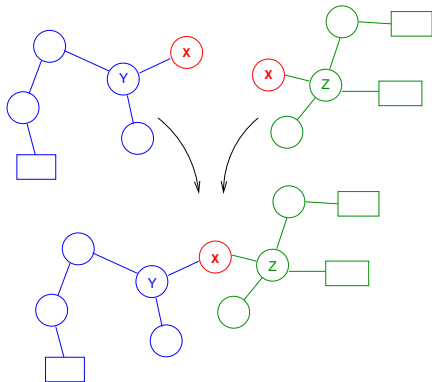
- Related information:
 - NAS & GRO: births, deaths, marriages – *Scotland's People*
 - RCAHMS: sites from Neolithic to Now – *Scotland's Places*
 - NMS: excavation finds, cultural objects
 - NLS: bibliographic material supporting all of it
- Interconnecting relational databases is hard:
 - you need to know the schema in detail
 - security issues
 - complex networking protocols – not http
 - whereas RDF was *designed* for data linking

Why Convert? – Interoperability

- Related information:
 - NAS & GRO: births, deaths, marriages – *Scotland's People*
 - RCAHMS: sites from Neolithic to Now – *Scotland's Places*
 - NMS: excavation finds, cultural objects
 - NLS: bibliographic material supporting all of it
- Interconnecting relational databases is hard:
 - you need to know the schema in detail
 - security issues
 - complex networking protocols – not http
 - whereas RDF was *designed* for data linking

Dataset Linking in RDF

- Same resource node appears in two graphs? *ie same URI*
 - graphs are automatically linked



Why Convert? – Access to Standard Vocabularies

- “Of course we believe in standards...
...that’s why we have so many to choose from.”
- Vocabulary / Upper Ontology / Thesaurus
- Lots being developed in RDF
 - VoCamp Galway 2008, 25th-26th Nov
 - Linking Open Data on the Semantic Web

Why Convert? – Access to Standard Vocabularies

- “Of course we believe in standards...
...that’s why we have so many to choose from.”
- Vocabulary / Upper Ontology / Thesaurus
- Lots being developed in RDF
 - VoCamp Galway 2008, 25th-26th Nov
 - Linking Open Data on the Semantic Web

Why Convert? – Access to Standard Vocabularies

- “Of course we believe in standards...
...that’s why we have so many to choose from.”
- Vocabulary / Upper Ontology / Thesaurus
- Lots being developed in RDF
 - [VoCamp Galway 2008](#), 25th-26th Nov
 - [Linking Open Data on the Semantic Web](#)

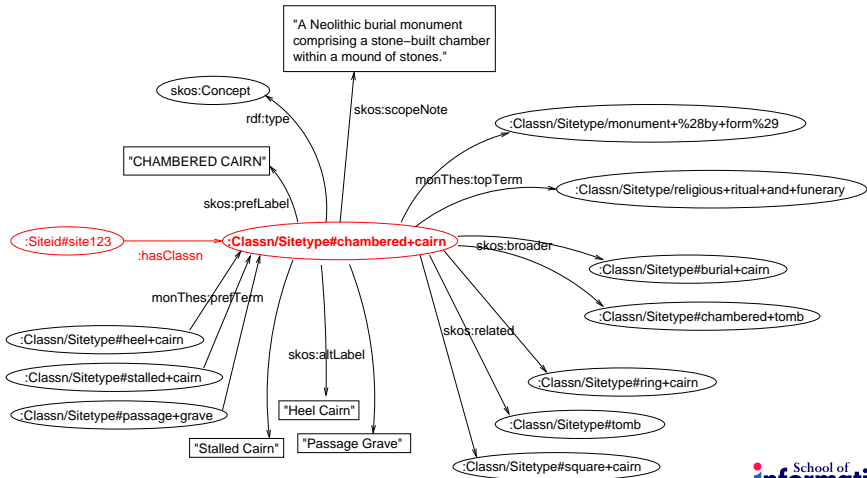
Standard Vocabularies for *Tether*

- **Monument Type Thesaurus** and **Object Type Thesaurus**
- Will link in turn to CIDOC-CRM (European standard)
- Well-structured data; easy to convert to RDF
- Used SKOS schema (Simple Knowledge Organisation System)
- Classification terms found in text automatically grounded
- Example: *site123* – *hasClassn* – “chambered cairn”

Term Grounding in *Tether*



Term Grounding in *Tether*



Summary

- Need to expose data as RDF to join Semantic Web
- RDF design needs as much care as relational database design
- *Tether* system integrates text with relational data
- Potentially big gains for interoperability...
- ...and for grounding data against standard ontologies