Publishing and Using Linked Data

Owen Stephens, November 2014

Using these slides

These slides were developed by Owen Stephens
(owen@ostephens.com).

URIs

URIs fundamental to Linked Data - can do RDF without creating/coining your own URIs but not really linked data “Use URIs as names for things”, “Use HTTP URIs so people can look up those names” (http://www.w3.org/DesignIssues/LinkedData)

http://
www.amazon.co.uk/
Pride-Prejudice-Penguin-
Classics-Austen/dp/
0141439513

What does this identify?

Doesn’t identify (as you might expect) Pride and Prejudice, but rather identifies the Amazon web page that describes the Penguin Classics edition of Pride and Prejudice. This may seem like splitting hairs, but if you want to start to make statements about things using their identifiers it is very important. I might want to state that the author of Pride and Prejudice is Jane Austen. If I say:

http://www.amazon.co.uk/Pride-Prejudice-Penguin-Classics-Austen/dp/0141439513 is authored by Jane Austen, then strictly I’m saying Jane Austen wrote the web page, rather than the book described by the web page.

http://data.open.ac.uk/page/person/
0e5d4257051894026ea74b7ed55557e7

http://data.open.ac.uk/person/
0e5d4257051894026ea74b7ed55557e7

Note the difference between these URIs - the first is about the page (click to show), the second is the ID for the person
One of the most hotly debated topics around Linked Data - three schools of thought:
* Use redirects
* Use URI fragments
* Accept and deal with the ambiguity

Cool URIs

TBL - “Cool URIs don’t change”
Avoid including specific technology and variables in your URIs
Some debate as to whether opaque or readable URIs are better - I suspect no right answer
To some extent we must also embrace (as on the web) “broken links aren’t the end of the world” - but they are a PITA

http://kmi.open.ac.uk/people.php?surname=daquin&forename=mathieu

http://data.open.ac.uk/person/daquin/mathieu

http://data.open.ac.uk/person/0e5d4257051894026ea74b7ed55557e7

First would be seen less ‘cool’ than second
Includes a department name (Knowledge Media Institute) – what if this changes it’s name or is disbanded?
Includes ‘php’ – what if the technology changes?
Includes field names – what if ‘surname’ changed to ‘family name’?
2nd better and is logically equivalent – webserver can manage translating one into the other
3rd is opaque – clear advantages (doesn’t matter if person changes their name) but disadvantages (unguessable/unreadable)

Choosing Vocabularies (...or Ontologies?)

Terms used somewhat interchangeably, although ‘ontology’ probably slightly more formal. Problem from library perspective is that ‘vocabulary’ in this context doesn’t mean a list of terms (like LCSH for example) but a schema for describing types of things

Shared ontologies are a way of agreeing we are describing the same type of thing.

To give a concrete example - Dublin Core is a widely used vocabulary. It contains a property of ‘creator’ (with a URI of http://purl.org/dc/terms/creator) - all users of this property in their data are working from a common definition (although this clearly doesn’t stop misuse!)

RDF

http://www.w3.org/1999/02/22-rdf-syntax-ns

Basic RDF syntax
Note especially rdf:type which is used to say what type of thing you are dealing with

RDFS

http://www.w3.org/2000/01/rdf-schema

RDF Schema - generally used for describing other vocabularies!
Note especially that rdfs:label turns up a lot in data and is often the place the actual literal string you are interested in is recorded

OWL

http://www.w3.org/2002/07/owl

The confusingly named ‘Web Ontology Language’ (working group quotes AA Milne in justification) – also underlying ontology that allows you to put constraints on other ontologies
But also the home of the infamous ‘sameAs’ statement which allows you to say one thing is the same as another thing in linked data

FOAF

http://xmlns.com/foaf/0.1

The increasingly inaccurately named ‘Friend of a Friend’ vocabulary – has become the defacto standard for describing people – names etc.

Dublin Core Terms

http://purl.org/dc/terms

Important to note this is not just the 15 elements that might spring to mind when you think of DC, although some of those are the most widely used - Title and Creator especially

SKOS

http://www.w3.org/2004/02/skos/core

Simple Knowledge Organization System
For any structured ‘vocabulary’ (not in RDF sense)
For example an entry in NAF would be a SKOS Concept. As would be any Authorized Library of Congress Subject Heading
SKOS has a ‘prefLabel’ property which is sometimes used as the ‘display’ label (as opposed to rdfs:label)

Other Vocabularies

- Bibliographic Ontology a.k.a BIBO (http://purl.org/ontology/bibo)
- Bio (http://vocab.org/bio/0.1/.html)
- FRBR (http://vocab.org/frbr/core.html)
- ISBD (http://iflastandards.info/ns/isbd)
- CRM (http://erlangen-crm.org/current)

All of these seeing some use in the ‘lodlam’ (linked open data in libraries, archives and museums) space. CRM being used extensively by the British Museum http://collection.britishmuseum.org)
... and more ... e.g. SPAR ontologies (http://sempublishing.sourceforge.net)

<http://data.lib.cam.ac.uk/id/entry/cambrdgedb_1000346> <http://purl.org/dc/terms/title> "Early medieval history of Kashmir";
<http://purl.org/dc/terms/identifier> "UkCU1000346";
<http://purl.org/dc/terms/language> <http://id.loc.gov/vocabulary/iso639-2/eng>;
<http://RDVocab.info/ElementsplaceOfPublication> <http://id.loc.gov/vocabulary/countries/ii>;
<http://iflastandards.info/ns/isbd/elements/P1016> "New Delhi".

Sample data (truncated) from Cambridge University Library
You can see that it uses DC, RDA and ISBD vocabularies/ontologies in single record description

Publishing Linked Data

Static files

Once you have data represented in RDF you can ‘publish’ this by simply putting the RDF file(s) in an accessible place on a web server.

This is like authoring ‘static’ html pages and uploading them - e.g. via ftp

Publishing Linked Data in this way is simple, but lacks sophistication and probably works better for small data sets - not for millions of triples that we might typically expect in library data - although there are probably arguments that much could be achieved by one RDF file per ‘record’

Large RDF files are sometimes published this way in conjunction with more sophisticated access to allow for easy download of large amount of data etc.

Example of publishing RDF as a simple static file: http://www.meanboyfriend.com/overdue_ideas/middlemash.rdf (for background see http://www.meanboyfriend.com/overdue_ideas/2009/10/middlemash-middlemarch-middlemap/)

Example of publishing large RDF dump: http://www.bl.uk/bibliographic/download.html#basicbnb

See also http://linkeddatabook.com/editions/1.0/#htoc66

Dynamically generated views

Typically based on data stored in a triple store (what's a triple store? basically a kind of database designed specifically to store RDF triples) or a more traditional database.

More like the way a blog like Wordpress works than a static html page

Software generates the 'views' (which can be HTML views of the data, or RDF in one or more serialisations, or other formats). The view you get is sometimes determined by the URL you use, or based on the type of request made to the web server which allows you to specify the format you want (this is called 'content negotiation' and is part of HTTP)

Lots of different ways of doing this - just as to publish HTML you can use Wordpress, Blogger, Drupal, other Content Mangement Systems etc. etc.

See:
http://linkeddatabook.com/editions/1.0/#htoc68
http://linkeddatabook.com/editions/1.0/#htoc69
http://linkeddatabook.com/editions/1.0/#htoc70
http://linkeddatabook.com/editions/1.0/#htoc71

Embedded in HTML

Publishing structured data embedded in HTML is becoming more common - this can be done with or without it being ‘linked data’. However there are ways of publishing ‘linked data’ in this way. The one that seems to have most momentum at the moment is ‘schema.org’ (http://schema.org) which is backed by Google/Yahoo/Yandex/Bing and others.

Whether ‘schema.org’ markup is linked data probably depends exactly how you use it. There is a mapping of schema.org to “RDFa Lite” which is an initiative from W3C (http://www.w3.org/TR/rdfa-lite/) to allow embedding of RDF in web pages

See
http://linkeddatabook.com/editions/1.0/#htoc67

Linking it up

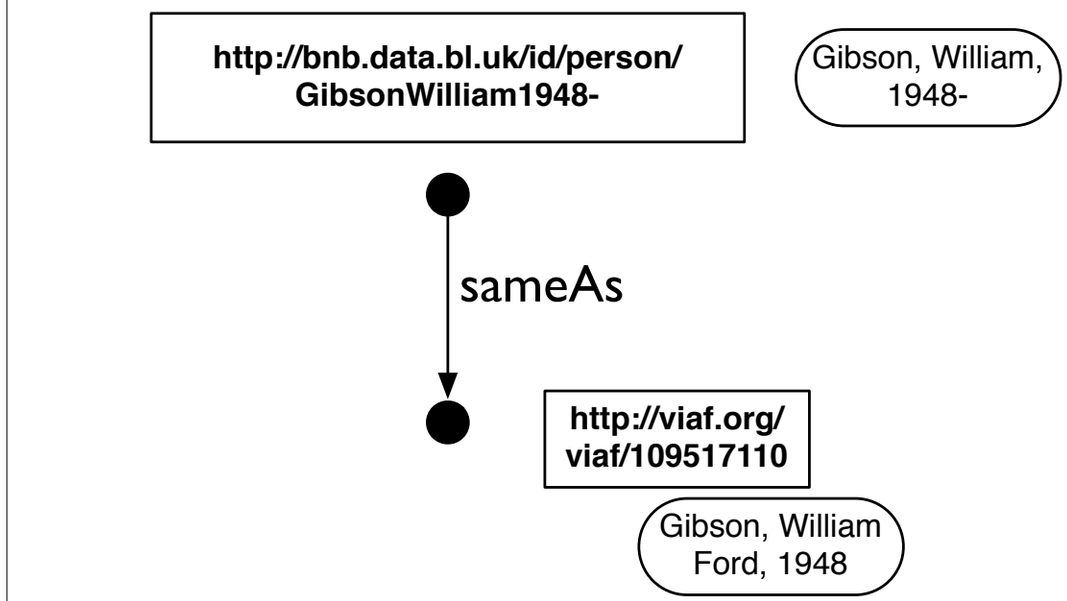
Key aspect of ‘linked data’ is ... the ‘links’.
Back to Linked Data design issues statement by TBL “Include links to other URIs. so that they can discover more things.” (http://www.w3.org/DesignIssues/LinkedData)

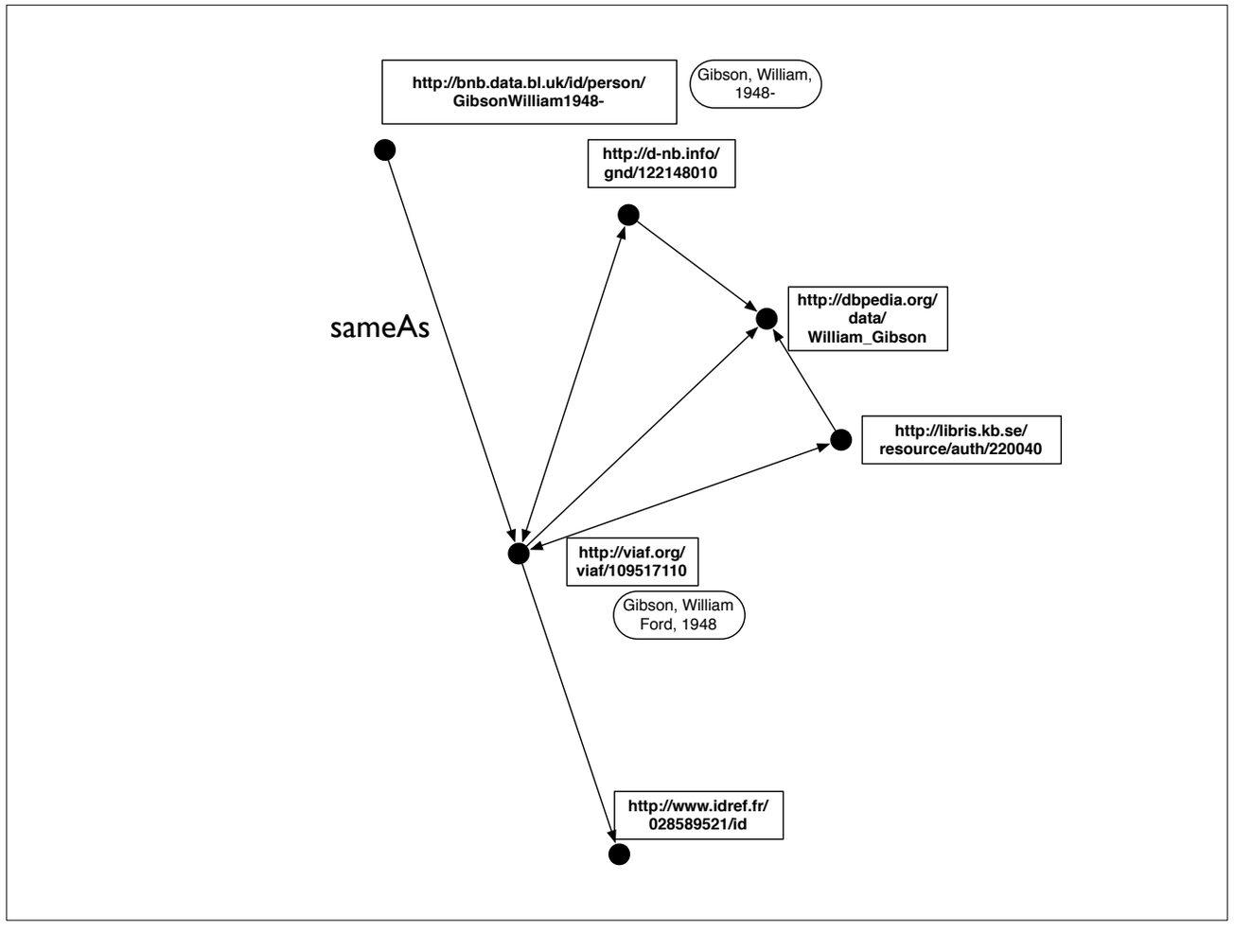
Typical to establish your own links first, then establish ‘sameAs’ statements to equivalent URIs elsewhere.

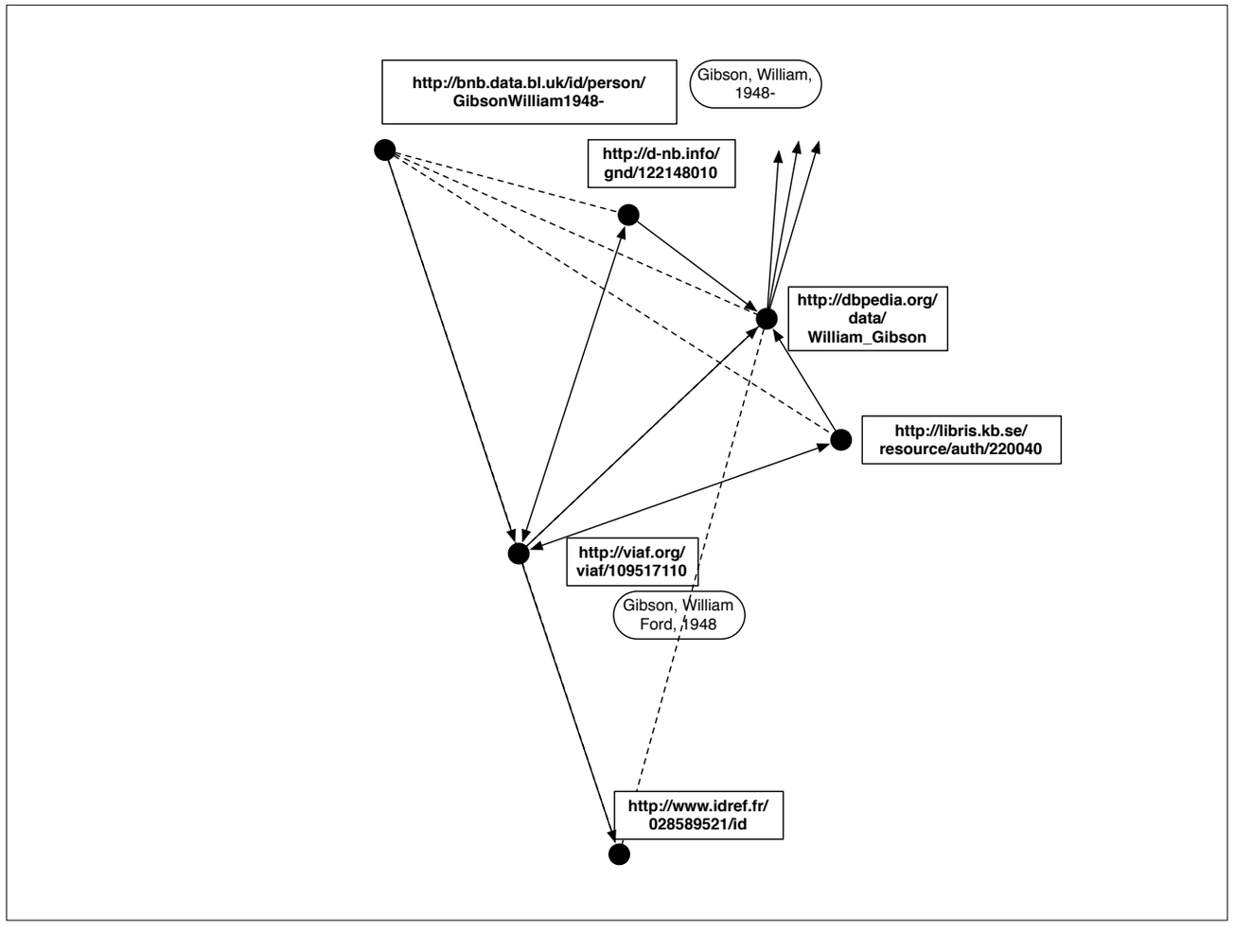
Not necessarily easy to establish equivalence between things (this is part of the point of moving to identifiers - to try to make this easier)

Sometimes can do this via existing identifiers
Sometimes need to do some lookup via text strings (e.g. with id.loc.gov this is an approach you can take)
Sometimes takes more work...
Tools that can help - OpenRefine (https://github.com/OpenRefine/OpenRefine/wiki) and SILK (http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/)


http://bnb.data.bl.uk/id/person/GibsonWilliam1948-
Gibson, William, 1948-


http://bnb.data.bl.uk/id/person/
GibsonWilliam1948-
Gibson, William,
1948-
sameAs
http://viaf.org/
viaf/109517110
Gibson, William
Ford, 1948


http://bnb.data.bl.uk/id/person/GibsonWilliam1948-
Gibson, William, 1948-
http://d-nb.info/gnd/122148010
sameAs
http://dbpedia.org/data/William_Gibson
http://libris.kb.se/resource/auth/220040
http://viaf.org/viaf/109517110
Gibson, William Ford, 1948
http://www.idref.fr/028589521/id


http://bnb.data.bl.uk/id/person/GibsonWilliam1948-
Gibson, William, 1948-
http://d-nb.info/gnd/122148010
http://dbpedia.org/data/William_Gibson
http://libris.kb.se/resource/auth/220040
http://viaf.org/viaf/109517110
Gibson, William Ford, 1948
http://www.idref.fr/028589521/id

Is it Open?

Not going to say a lot about this but ... what licence is used in it's publication?
BL published BNB under CC0 declaration - puts it in the public domain
Archives Hub did the same
Cambridge ended up with a mixture of 'attribution' licence (ODC-BY) and public domain (ODC-PDDL) - depending on the rights they had over their metadata (see http://data.lib.cam.ac.uk/datasets.php)
Europeana goes with CC0
DPLA says 'we don't believe copyright applies to metadata ... but if it does then we put it in the public domain' (http://dp.la/info/wp-content/uploads/2013/04/DPLAMetadataPolicy.pdf)

Consuming Linked Data

Crawling

“due to the likelihood of scalability problems with on-the-fly link traversal and federated querying, it may transpire that widespread crawling and caching will become the norm in making data from a large number of data sources available”

If we envisage a distributed bibliographic data environment, it is this approach we’d need to take to build the equivalent of an OPAC.

http://linkeddatabook.com/editions/1.0/#htoc84

This type of approach is what Google does! See also the ‘CommonCrawl’ http://commoncrawl.org which is trying to make a public version of a web crawl available - the equivalent could be done for linked data (or they may turn out to be the same thing)

Definitely a challenging area - see my blog post http://www.meanboyfriend.com/overdue_ideas/2012/08/what-to-do-with-linked-data/

Follow your nose & Just in time

One of the key aspects of linked data is that the links enable to take a ‘follow your nose’ approach to the available links. For some applications this is all that is needed

For example I wrote a bookmarklet (a way of adding functionality/data to websites by clicking a browser bookmark) using this approach - see http://www.meanboyfriend.com/overdue_ideas/2011/07/compose-yourself/

It works as long as you can return enough information quickly enough. It wouldn’t work for (e.g.) a search application where you can’t really index all the relevant information ‘just in time’, but it could work where you wanted to enhance the display of a record in an OPAC (or equivalent) ‘just in time’ when a person views the record.

Federated Queries

Idea being send out queries to distributed linked data sources using SPARQL (a query language for RDF Triple stores).... anyone who has dealt with federated search in libraries will know the challenges that this can bring!

http://linkeddatabook.com/editions/1.0/#htoc84

Further Reading

Linked Data Design Issues:
http://www.w3.org/DesignIssues/LinkedData

The Linked Data book
http://linkeddatabook.com/editions/1.0/

Further thoughts on using linked data from me
http://www.meanboyfriend.com/overdue_ideas/2012/08/what-to-do-with-linked-data/

Publishing and Using Linked Data

Owen Stephens, November 2014