draft Draft Draft! DRAFT! D R A F T ! ! !

LINKED ARCHIVAL METADATA

A GUIDEBOOK

Eric Lease Morgan
January 18, 2014

Executive Summary

[The Executive Summary will list core objectives, anticipated outcomes, and implications that will provide administrators or other senior leaders with the information that they will need in order to understand the benefits and potential costs of this path.]

Introduction

Linked Archival Metadata: A Guidebook provides archivists with an overview of the current linked data landscape, define basic concepts, identify practical strategies for adoption, and emphasize the tangible payoffs for archives implementing linked data. It focuses on clarifying why archives and archival users can benefit from linked data and will identify a graduated approach to applying linked data methods to archival description.

The Guidebook is a product of the Linked Archival Metadata planning project (LiAM), led by the Digital Collections and Archives at Tufts University and funded by the Institute of Museum and Library Services (IMLS). LiAM's goals include defining use cases for linked data in archives and providing a roadmap to describe options for archivists intending to share their description using linked data techniques.

Why linked data, and why now?

Linked data, or more recently referred to as "linked open data" for reasons to be explained later, is a proposed technique for generating new knowledge. It is intended to be a synergy between people and sets of agreed upon computer systems that when combined will enable both people and computers to discover and build relationships between seemingly disparate data and information to create and discover new knowledge.

In a nutshell, this is how it works. People possess data and information. They encode that data and information in any number of formats easily readable by computers. They then make the encoded data and information available on the Web. Computers are then employed to systematically harvested the encoded data. Since the data is easily readable, the computers store the data locally and look for similarly encoded things in other locally stored data sets. When similar items are identified relationships can be inferred between the items as well as the other items in the data set. To people, some of these relationships may seem obvious and "old hat". On the other hand, since the data sets can be massive, relationships that were never observed previously may come to light, thus new knowledge is created.

Some of this knowledge may be trivial. For example, there might be a data set of places -- places from all over the world including things like geographic coordinates, histories of the places, images, etc. There might be another data set of poeple. Each person may be described using their name, their place of birth, and a short biography. These data sets may contain ten's of thousands of items each. Using linked data it would be possible to cross reference the people with the places to discover who might have met whom when and where. Some people may have similar ideas, and those ideas may have been generated in a particular place. Linked data may help in discovering who was in the same place at the same time and the researcher may be better able to figure out how a particular idea came to fruition.

Here's an example hitting closer to the home of archives and archivists. Suppose most archival finding aids were written in a format easily readable by computers. Let's call this format Encoded Archival Description. Let's suppose these finding aids were made available on the Web. Let's suppose one or more computers crawled these archival sites harvesting the finding aids. Once done a computer program could be used to find all the occurrences of particular name and generate a virtual finding aid that is more complete and more comprehensible than any single finding aid on that particular person.

The amount of data and information accessible today is greater in size than it has ever been in human history. Using our traditional techniques of reading, re-reading, writing, discussing, etc. is more than possible to learn new things about the state of the world, the universe, and the human condition. By exploiting the current state of computer technology is possible to expand upon our traditional techniques and possibly accelerate the mass of knowledge.

How to use the Guidebook

The structure of the Guidebook supports readers moving through the text in a variety of ways. Like a travel book, it provides useful high-level information for users who only need the basics, as well as in-depth information for those planning an extended stay in LOD-land. The Guidebook is intentionally named, and will draw from the genre of actual travel guides (Fodors, etc.) providing readers easy access to both high-level

information (know before you go, what to see if you're only there for a day) as well as in-depth details of for those staying in one place longer.

Synopses of the use cases developed by the LiAM project will be interspersed throughout the Guidebook to illustrate and frame the text. Each use case will be briefly described in 100-200 words with links to the full use cases on the LiAM website.

An initial release of the Guidebook will be in the form of a PDF document to be delivered to IMLS in fulfillment of the LiAM planning grant requirements as well as being shared with the public. However, the Guidebook's ongoing vitality will benefit from a more dynamic publication environment, and we therefore plan to publish it in a wiki connected to a code repository. This combination will enable updating of the resource to reflect changes in the field as well as providing a mechanism for sharing tools, scripts, and other code related to the project.

Much of the rest of the Guidebook, while providing a concise overview of today's linked data landscape and needs, would require ongoing updates, maintenance, and enhancement to describe implementation of LOD in the archival community over time.

Linked Data for Archives: a Primer

Linked Data is a process for manifesting the ideas behind Semantic Web. The Semantic Web is about encoding data, information, and knowledge in computer-readable fashions, making these encodings accessible on the World Wide Web, allowing computers to crawl the encodings, and finally, employing reasoning engines against them for the purpose of discovering and creating new knowledge. The canonical article describing this concept was written by Tim Berners-Lee, James Hendler, and Ora Lassila in 2001.

In 2006 Berners-Lee more concretely described how to make the Semantic Web a reality in a text called “Linked Data -- Design Issues”. In it he outlined four often-quoted expectations for implementing the Semantic Web. Each of these expectations are listed below along with some of my own elaborations:

1. “Use URIs as names for things” - URIs (Universal Resource Identifiers) are unique identifiers, and they are expected to have the same shape as URLs (Universal Resource Locators). These identifiers are expected to represent things such as people, places, institutions, concepts, books, etc. URIs are monikers or handles for real world or imaginary objects.

2. “Use HTTP URIs so that people can look up those names.” - The URIs are expected to look and ideally function on the World Wide Web through the Hypertext Transfer Protocol (HTTP), meaning the URI’s point to things on Web servers.

3. “When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL)” - When URIs are sent to Web servers by Web browsers (or “user-agents” in HTTP parlance), the response from the server should be in a conventional, computer readable format. This format is usually a “serialization” of RDF (Resource Description Framework) -- a notation looking much like a rudimentary sentence composed of a subject, predicate, and object.

4. “Include links to other URIs. So that they can discover more things.” - Simply put, try very hard to use URIs that other people have have used. This way the relationships you create can literally be linked to the relationships other people have created. These links may represent new knowledge.

In the same text (“Linked Data -- Design Issues”) Berners-Lee also outlined a sort of reward system -- sets of stars -- for levels of implementation. Unfortunately, nobody seems to have taken up the stars very seriously. A person gets:

- 1 star for making data available on the web (in whatever format) but with an open license, to be Open Data
- 2 stars for making the data machine-readable structured data (e.g. excel instead of image scan of a table)
- 3 stars for making the data available in non-proprietary format (e.g. CSV instead of excel)
- 4 stars for using open standards from W3C (RDF and SPARQL) to identify things, so that people can point at your stuff
- 5 stars for linking your data to other people’s data to provide context

The whole idea works like this. Suppose I assert the following statement:

The Declaration Of Independence was authored by Thomas Jefferson.

This statement can be divided into three parts. The first part is a subject (Declaration Of Independence). The second part is a predicate (was authored by). The third part is an object (Thomas Jefferson). In the language of the Semantic Web and Linked Data, these combined parts are called a triple, and they are expected to denote a fact. Triples are the heart of RDF.

Suppose further that the subject and object of the triple are identified using URIs (as in Expectations #1 and #2, above). This would turn our assertion into something like this with carriage returns added for readability:

> http://en.wikipedia.org/wiki/Declaration_of_Independence
> was authored by
> http://www.worldcat.org/identities/lccn-n79-89957

Unfortunately, this assertion is not easily read by a computer. Believe it or not, something like the XML below is much more amenable, and if it were the sort of content returned by a Web server to a Web browser (read “user-agent”), then it would satisfy Expectations #3 and #4 because the notation is standardized and because it points to other people’s content:

```
<?xml version=”1.0”?>
<rdf:RDF
  xmlns:rdf=”http://www.w3.org/1999/02/22-rdf-syntax-ns#”
  xmlns:dcterms=”http://purl.org/dc/terms/” >

  <!-- the Declaration Of Independence was authored by Thomas Jefferson -->
  <rdf:Description
  rdf:about=”http://en.wikipedia.org/wiki/Declaration_of_Independence”>
    <dcterms:creator>http://id.loc.gov/authorities/names/n79089957</dcterms:creator>
  </rdf:Description>

</rdf:RDF>
```

Suppose we had a second assertion:

> Thomas Jefferson was a man.

In this case, the subject is “Thomas Jefferson”. The predicate is “was”. The object is “man”. This assertion can be expressed in a more computer-readable fashion like this:

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <!-- Thomas Jefferson is man (a male) -->
  <rdf:Description rdf:about="http://id.loc.gov/authorities/names/n7908995">
    <foaf:Person foaf:gender="male" />
  </rdf:Description>

</rdf:RDF>
```

Suppose there were smart Linked Data robot / spider. Suppose it crawled both Assertion #1 and Assertion #2, it then ought to be able to assert the following:

```
<?xml version="1.0"?>
<rdf:RDF
 xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <!-- the Declaration Of Independence was written by
  Thomas Jefferson, and Thomas Jefferson is a male -->
  <rdf:Description rdf:about="http://en.wikipedia.org/wiki/
Declaration_of_Independence">
 <dcterms:creator>
 <foaf:Person rdf:about="http://id.loc.gov/authorities/names/n79089957">
 <foaf:gender>male</foaf:gender>
 </foaf:Person>
 </dcterms:creator>
  </rdf:Description>

</rdf:RDF>
```

Looking at the two assertions, a reasonable person can deduce a third assertion, namely, the Declaration Of Independence was authored by a man. Which brings us back to the point of the Semantic Web and Linked Data. If everybody uses URIs (read "URLs") to describe things, if everybody denotes relationships (through the use of predicates) between URIs, if everybody makes their data available on the Web in standardized formats, and if everybody uses similar URIs, then new knowledge can be deduced from the original relationships.

Unfortunately and to-date too little Linked Data has been made available and/or too few people have earned too few stars to really make the Semantic Web a reality. The purpose of this guidebook is to provide means for archivists to do their part, make their content available on the Semantic Web through Linked Data, all in the hopes of facilitating the discovery of new knowledge. On our mark. Get set. Go!

There are a number of challenges in the process. Some of them are listed below, and some of them have been alluded to above:

Create useful LOD, meaning, create LOD that links to other LOD. LOD does not live in a world by itself. Remember, the "L" stands for "linked". For example, try to include URIs that are the URIs used on other LOD data sets. Sometimes this is not possible, for example, le with the names of people in archival materials. When possible, they used VIAF, but other times they needed to create their own URI denoting an individual.

There is a level of rigor involved in creating the data model, and there may be many discussions regarding semantics. For example, what is a creator? Or, when is a term intended to be an index term as opposed reference. When does one term in one vocabulary equal a different term in a different vocabulary?

Balance the creation of your own vocabulary with the need to speak the language of others using their vocabulary.

Consider "fixing" the data as it comes in or goes out because it might not be consistent nor thorough.

Provenance is an issue. People — especially scholars — will want to know where the LOD came from and whether or not it is authoritative. How to solve or address this problem? The jury is still out on this one.

Creating and maintaining LOD is difficult because it requires the skills of a number of different types of people. Computer programmers. Database designers. Subject experts. Metadata specialists. Archivists. Etc. A team is all but necessary.

Objectives

[Management, access, and use and linked data affordances]

Overview of linked data concepts and vocabulary

Linked Data is a process for systematically and methodically exposing metadata on the Web. In many ways, it is the re-articulation of a thing called the Semantic Web first outlined more than a decade ago. Linked Data (and the Semantic Web) are efforts to increase the “sphere of knowledge” through the use of computer technology.

Increasingly you will hear of of linked data being qualified as “linked open data”. The “open” qualifier alludes to the important distinctions between truly free data/information and licensed data/information which comes with some strings attached. Truly “open” linked data comes with no licensing restrictions.

When you hear of linked data and the Semantic Web, the next thing you often hear is “RDF” or “Resource Description Framework”. First and foremost, RDF is a way of representing knowledge. It does this through the use of assertions (think, “sentences”) with only three parts: 1) a subject, 2) a predicate, and 3) an object. Put together, these three things create things called “triples”. The subject of each assertion is expected to be a Universal Resource Identifier (or URI, but think URL), and this URI is expected to represent a thing -- anything. (Really, anything.) The predicate is some sort of relationship such as equals or is a sub-part of or contains or is a description of, or is the name of, etc. Predicates are the vocabulary of linked data, and you will find an abundance of vocabularies from which to choose when creating Linked Data. Finally,

objects come in two forms: 1) more URIs (pointers to things) or literal values such the names of people, places, or things. Examples of literals include “Lancaster, PA”, “Thomas Jefferson”, or “Musée d’Orsay”.

RDF is not to be confused with RDF/XML or another other type of RDF “serialization”. Remember, RDF describes triples, but it does not specify how the triples are express or written down. On the other hand, RDF/XML is an XML syntax for expressing RDF. Some people think RDF/XML is too complicated and too verbose. Consequently, other serializations have manifested themselves including N3 and Turtle.

Brief overview of the history of LOD-LAM

Examples

Linked Data Today

Projects

[Brief descriptions with an emphasis on tangible benefits and outcomes of each]

Google (and Facebook) knowledge graphs

OpenCat

> Another common theme / application demonstrated at the conference were variations of the venerable library catalog. OpenCat, presented by Agnes Simon (Bibliothéque Nationale de France), was an additional example of this trend. Combining authority data (available as RDF) provided by the National Library of France with works of a second library (Fresnes Public Library), the OpenCat prototype provides quite an interesting interface to library holdings. --http://demo.cubicweb.org/opencatfresnes/

Newspaper Clippings Archives

> On Jan 8, 2014, at 12:05 PM, Neubert Joachim <J.Neubert@zbw.eu> wrote:
>
> Thank you for your report on SWIB13! I'm glad you enjoyed the conference and your stay in Hamburg.
>
> I wanted to get in touch with you because you mentioned in your blog that you are working on a book about LOD in archives. Perhaps, in your research for that, you came across press/newspaper clippings archives.
>
> As you may know, I've published the persons and company part of the 20th Century Press Archives (http://zbw.eu/beta/p20) as a linked data application. It uses RDFa and OAI-ORE extensively to

give every dossier, every article and every page a citable URI, and on the other hand consumes linked data from various linked data sources to enrich the web pages and to provide context to the rather plain scanned article images.

I wonder if there are other archives, and particular newspaper archives, out there which do similar things, and would be very happy about hints.

http://challenge.semanticweb.org/submissions/swc2010_submission_6.pdf
http://elag2011.techlib.cz/files/download/id/45/drawing-context-from-the-linked-data-web-the-20th-century-press-archives-neubert.pdf
http://www.w3.org/2005/Incubator/lld/wiki/Use_Case_Publishing_20th_Century_Press_Archives

--
Joachim Neubert
ZBW – German National Library of Economics
Leibniz Information Centre for Economics
Neuer Jungfernstieg 21
20354 Hamburg

LOCAH Project

Mimas and UKOLN worked together on an exciting JISC funded project to make Archives Hub data available as structured Linked Data, for the benefit of education and research. We worked in partnership with Eduserv, Talis and OCLC, leading experts within their fields. The aim was put archival and bibliographic data at the heart of the Linked Data Web, enabling new links to be made between diverse content sources and enabling the free and flexible exploration of data so that researchers can make new connections between subjects, people, organisations and places to reveal more about our history and society. --http://archiveshub.ac.uk/locah/

Linking Lives

Linking Lives is exploring ways to present Linked Data. We aim to show that archives can benefit from being presented as a part of the diverse data sources on the Web to create full biographical pictures, enabling researchers to make connections between people and events.

Linking Lives builds upon the Locah project. Locah was a JISC-funded project to expose the Archives Hub descriptions as Linked Data. --http://archiveshub.ac.uk/linkinglives/

Linked Archives Hub Test Dataset

The dataset describes archives held by UK institutions. The data is derived from a sample of the archival finding aids held by the UK Archives Hub. --http://data.archiveshub.ac.uk

Trends in LOD-LAM

Mash ups
Harvesting along side other protocols
Increased interest
Increased number of RDF serializations
Governments making their content available
Using them to enhance online catalogs
Creating timelines
Creating “named graphs”
Increased number of programming toolkits
Emphasis on “open” linked data and linked data in museums and archives
Making RDF dumps available
Interest in schema.org

With great interest I read the Spring/Summer issue of Information Standards Quarterly where there were a number of articles pertaining to linked open data in cultural heritage institutions. [0] Of particular interest to me where the various loosely enumerated challenges of linked open data. Some of them included:

- the apparent Tower Of Babel when it comes to vocabularies used to describe content, and the same time we need to have “ontology mindfulness”.

- dirty, inconsistent, or wide varieties of data integrity

- persistent URIs

- the “chicken & egg” problem of why linked data if there is no killer application

Getting Started: Strategies and Steps

Defining your strategy

Linked data represents a modern way of making your archival descriptions accessible to the wider world. In that light, it represents a different way of doing things but not necessary a different what of doing things. You will still be doing inventory. You will still be curating collections. You will still be prioritizing what goes and what stays.

On the other hand, linked data changes the way your descriptions get expressed and distributed. It is a lot like taking a trip across country. The goal was always to get to the coast to see the ocean, but instead of walking, going by stage coach, taking a train, or driving a car, you will be flying. Along the way you may visit a few cities and have a few layovers. Bad weather may even get in the way, but sooner or later you will get to your destination. Take a deep breath. Understand that the process will be one of learning, and that learning will be applicable in other aspects of your work. The result will be two-fold. First, a greater number of people will have access to your collections, and consequently, more people will will be using your collections.

With this in mind, articulate some goals — broad targets of things you would like to accomplish. Some of them might include:

- making your archival collections more widely accessible
- working with others to build virtual collections of like topics or formats
- incorporating your archival descriptions into public spaces like Wikipedia
- integrating your collections into local teaching, learning, and research activities
- increasing the awareness of your archive to benefactors
- increasing the computer technology skills of fellow archivists

How might you go about accomplishing these goals? What are your objectives? (What method of transportation are you going to use to get where you are going?) How am I going to measure success? In other words, you will need to create an plan, and each item in the plan answers a simple question — Who is going to do what by when? In

other word, what people will be responsible for accomplishing the particular objective. Exactly what will they be doing, and by what time will they have it accomplished. Each of these components are described in greater detail below

Who

It is quite unlikely your linked data goals and objectives will be accomplished by a single person. Instead it will most likely required a team of people. These people do not necessarily need to working in the same physical location, but they will require a diverse set of skills. Some of them include, and each plays a key, indispensable role:

content specialists - These are the people who understand the “aboutness” of a particular collection. These are the people who understand and can thoroughly articulate the significance of a collection. They know how and why particular things belong in a collection. They are able to answer questions about the collection as all as tell stories against it.

metadata specialists - These are people who understand data about data. Not only do they understand the principles of controlled vocabularies and authority lists, but they are also familiar with a wide variety of such lists, specifically as they are represented on the Web. In linked data there are fewer descriptive cataloging “rules”. Nevertheless, the way the ontologies of linked data can be used need to be interpreted, and this interpretation needs to be consistent. Metadata specialists understand these principles.

computer technologists - Not only are these the people who have a fundamental understanding of what computer can and cannot do, but they also know how to put this understanding into practice. At the very least, the computer technologists need to understand a myriad of data structures and how to convert them into different data structures. Converting MARC 21 into MARCXML. Transforming EAD into HTML. Reporting against a relational database to create serialized RDF. These tasks required computer programming skills, but not necessarily any one in particular. Any modern programming language (Java, PHP, Python, Ruby, etc.) includes the necessary function to complete the tasks.

What

The what of your objectives are not so much identified with nouns as they are action verbs, such as: write, evaluate, implement, examine, purchase, hire, prioritize, list, delete, acquire, discuss, share, find, compare & contrast, stop, start, complete, continue, describe, edit, updated, create, purchase, upgrade, etc. The what of your objective is in the doing.

When

The say, “Work expands to fill the available space.” If this is true, and no deadlines are articulated for each objective, then the allotted amount of time for any given task is all but infinite, but this it not true. Time is one of the most limited resources you have. When thinking about a given objective, ask yourself how much time you think it will take, multiply the time by one and a half. Ask yourself when the task can begin and document the beginning point as well as the estimated ending point. Do this all of your objectives and the result will be a Gantt chart. It will now be easy to look at the chart on a regular basis to see who things are progressing.

[Articulate goals, objectives, and metrics to measure success.]

Is your archival description LOD-ready?

Identify building blocks

The building blocks of linked data include:

- URIs pointing to real-world objects: people, places, or things where things can be ideas or just about anything on the Web
- Ontologies, the language(s) of relationships between the URIs
- Content to share with the wider world

- People to do the work

- Computer technology to manifest the work

Probably one of the more difficult intellectual tasks you will have when it comes to making your content available as linked data will be the selection of one or more ontologies used to make relationship between the subjects and objects of your triples. Probably the easiest way to think about these ontologies is as if they were fields in a MARC record or EAD file. Such an analogy is useful, but not 100% correct. Probably the best way to think of the ontologies is as if they were verbs in a sentence denoting relationships between things — subjects and objects. But if ontologies are sets of "verbs", then they are akin to human language, and human language is ambiguous. Therein lies the difficulty with ontologies. There is no "right" way to implement them. Instead, there is only best or common practice. There are no hard and fast rules. Everything comes with a bit of interpretation. The application and use of ontologies is very much like the application and use of written language in general. In order for written language to work well two equally important things need to happen. First, the writer needs to be able to write. They need to be able to choose the most appropriate language for their intended audience. Shakespeare is not "right" with his descriptions of love, but instead his descriptions of love (and many other human emotions) resinate with a very large number of people. Second, written language requires the reader to have a particular adeptness as well. Shakespeare can not be expected to write one thing and communicate to everybody. The reader needs to understand English, or the translation from English into another language needs to be compete and accurate.

The Internet, by design, is a decentralized environment. There are very few rules on how it is expected to be used. To a great extent it relies on sets of behavior that are more common practice as opposed to articulated rules. For example, what "rules" exist for tweets on Twitter? What rules exist for Facebook or blog postings. Creating sets of rules will not fly on the Internet because there is no over-arching governing body to enforce any rules. Sure, there are things like Dublin Core with their definitions, but those definitions are left to interpretation, and there are no judges nor courts nor laws determining whether or not any particular application of Dublin Core is "correct". Only the common use of Dublin Core is correct, and its use is not set in stone.

There are no "should's" on the Internet. There is only common practice.

With this in mind, it is best for you to work with others both inside and outside your discipline to select one or more ontologies to be used in your linked data. Do not think about this too long nor too hard. It is an never-ending process that is never correct. It is only a process that approximates the best solution.

For simplicity's sake, RDF ontologies are akin to the fields in MARC records or the entities in EAD/XML files. Articulated more accurately, they are the things denoting relationships between subjects and objects in RDF triples. In this light, they are akin to the verbs in all but the most simplistic of sentences. But if they are akin to verbs, then they bring with them all of the nuance and subtlety of human written language. And human written language, in order to be an effective human communications device, comes with two equally important prerequisites: 1) a writer who can speak to an intended audience, and 2) a reader with a certain level of intelligence. A writer who does not use the language of the intended audience speaks to few, and a reader who does not "bring something to the party" goes away with litte understanding. Because the effectiveness of every writer is not perfect, and because not every reader comes to the party with a certain level of understanding, written language is imperfect. Similarly, the ontologies of linked data are imperfect. There are no perfect ontologies nor absolutely correct uses of them. There are only best practices and common usages.

This being the case, ontologies still need to be selected in order for linked data to be manifested. What ontologies would you suggest be used when creating linked data for archival descriptions? Here are a few possibilities, listed in no priority order:

* Dublin Core Terms - This ontology is rather bibliographic in nature, and provides a decent framework for describing much of the content of archival descriptions.

* FOAF - Archival collections often originate from individual people. Such is the scope of FOAF, and FOAF is used by a number of other sets of linked data.

* Schema.org - This is an up-and-coming ontology heralded by the 600-pound gorillas in the room -- Google, Microsoft, Yahoo, etc. While the ontology has not been put into practice for very long, it is growing and wide ranging.

* RDF - This ontology is necessary because linked data is manifested as... RDF

* RDFS - This ontology may be necessary because the archival community may be creating some of its own ontologies.

* OWL and SKOS - Both of these ontologies seem to be used to denote relationships between terms in other ontologies. In this way they are used to create classification schemes and thesauri. For example, they allow the implementor to that "creator" in one ontology is the same as "author" in another ontology. Or they allow "country" in one ontology to be denoted as a parent geographic term for "city" in another ontology.

While some or all of these ontologies may be useful for linked data of archival descriptions, what might some other ontologies include? (Remember, it is often "better" to select existing ontologies rather than inventing, unless there is something distinctly unique about a particular domain.) For example, how about an ontology denoting times? Or how about one for places? FOAF is good for people, but what about organizations or institutions?

[metadata components in archival description that are (or nearly are) ready for linking.]

Readiness

[Making small changes in practice to make your description LOD-ready.]

What you can do now if you have

Nothing - consider using RDFa

EAD

If you have used EAD to describe your collections, then you can easily make your descriptions available as valid linked data, but the result will be less than optimal. This is true not for a lack of technology but rather from the inherent purpose and structure of EAD files.

A few years ago an organisation in the United Kingdom called the Archive's Hub was funded by a granting agency called JISC to explore the publishing of archival descriptions as linked data. One of the outcomes of this effort was the creation of an XSL stylesheet transforming EAD into RDF/XML. The terms used in the stylesheet originate from quite a number of standardized, widely accepted ontologies, and with only the tiniest bit configuration / customization the stylesheet can transform a generic EAD file into valid RDF/XML. The resulting XML files can then be made available on a Web server or incorporated into a triple store. This goes a long way to publishing archival descriptions as linked data. The only additional things needed are a transformation of EAD into HTML and the configuration of a Web server to do content-negotiation between the XML and HTML.

For the smaller archive with only a few hundred EAD files whose content does not change very quickly, this is a simple, feasible, and practical solution to publishing archival descriptions as linked data. With the exception of doing some content-negotiation, this solution does not require any computer technology that is not already being used in archives, and it only requires a few small tweaks to a given workflow:

1. implement a content-negotiation solution
2. edit EAD file
3. transform EAD into RDF/XML
4. transform EAD into HTML
5. save the resulting XML and HTML files on a Web server
6. go to step #2

On the other hand an EAD file is the combination of a narrative description with a hierarchal inventory list, and this data structure does not lend itself very well to the triples of linked data. For example, EAD headers are full of controlled vocabularies terms but there is no way to link these terms with specific inventory items. This is because the vocabulary terms are expected to describe the collection as a whole, not

individual things. This problem could be overcome if each individual component of the EAD were associated with controlled vocabulary terms, but this would significantly increase the amount of work needed to create the EAD files in the first place.

The common practice of using literals (“strings”) to denote the names of people, places, and things in EAD files would also need to be changed in order to fully realize the vision of linked data. Specifically, it would be necessary for archivists to supplement their EAD files with commonly used URIs denoting subject headings and named authorities. These URIs could be inserted into id attributes throughout an EAD file, and the resulting RDF would be more linkable, but the labor to do so would increase, especially since many of the named authorities will not exist in standardized authority lists.

Despite these short comings, transforming EAD files into some sort of serialized RDF goes a long way towards publishing archival descriptions as linked data. This particular process is a good beginning and outputs valid information, just information that is not as accurate as possible. This process lends itself to iterative improvements, and outputting something is better than outputting nothing. But this particular proces is not for everybody. The archive whose content changes quickly, the archive with copious numbers of collections, or the archive wishing to publish the most accurate linked data possible will probably not want to use EAD files as the root of their publishing system. Instead some sort of database application is probably the best solution.

EAC-CPF

Encoded Archival Context for Corporate Bodies, Persons, and Families (EAC-CPF) goes a long way to implementing a named authority database that could be linked from archival descriptions. These XML files could easily be transformed into serialized RDF and therefore linked data. The resulting URIs could then be incorporated into archival descriptions making them richer and complete.

For example the FindAndConnect site in Australia uses EAC-CPF under the hood to disseminate information about people in its collection -- http://www.findandconnect.gov.au. Similarly, “SNAC aims to not only make the [EAC-CPF] records more easily discovered and accessed but also, and at the same time, build an

unprecedented resource that provides access to the socio-historical contexts (which includes people, families, and corporate bodies) in which the records were created" -- http://socialarchive.iath.virginia.edu More than a thousand EAC-CPF records are available from the RAMP project -- http://demo.rampeditor.info/export.php

MARC

In some ways MARC lends it self very well to being published via linked data, but in the long run it is not really a feasible data structure.

Converting MARC into serialized RDF through XSLT is at least a two step process. The first step is to convert MARC into MARCXML. This can be done with any number of scripting languages and toolboxes. The second step is to use a stylesheet such as the one provided by the Library of Congress to transform the MARCXML into RDF/XML. From there a person could save the resulting XML files on a Web server, enhance access via content negotiation, and called it linked data.

Unfortunately, this particular approach has a number of drawbacks. First and foremost, the MARC format had no place to denote URIs; MARC records are made up almost entirely of literals. Sure, URIs can be constructed from various control numbers, but things like authors, titles, subject headings, and added entries will most certainly be strings ("Mark Twain", "Adventures of Huckleberry Finn", "Bildungsroman", or "Samuel Clemans"), not URIs. This issue can be overcome if the MARCXML were first converted into MODS and URIs were inserted into id or xlink attributes of bibliographic elements, but this is extra work. If an archive were to take this approach, then it would also behoove them to use MODS as their data structure of choice, not MARC. Continually converting from MARC to MARCXML to MODS would be expensive in terms of time. Moreover, with each new conversion the URIs from previous iterations would need to be re-created.

METS and MODS

If you have archival descriptions in either of the METS or MODS formats, then transforming them into RDF is as far away as your XSLT processor and a content negotiation implementation. As of this writing there do not seem to be any METS to

RDF stylesheets, but there are a couple stylesheets for MODS. The biggest issue with these sorts of implementations are the URIs. It will be necessary for archivists to include URIs into as many MODS id or xlink attributes as possible. The same thing holds true for METS files except the id attribute is not designed to hold external identifiers and therefore not a valid placeholder for URIs.

Databases

Publishing linked data through XML transformation is functional but not optimal. Publishing linked data from a database comes closer to the ideal but requires a greater amount of technical computer infrastructure and expertise.

Databases -- specifically, relational databases -- are the current best practice for organizing data. As you may or may not know, relational databases are made up of many tables of data joined with keys. For example, a book may be assigned a unique identifier. The book has many characteristics such as a title, number of pages, size, descriptive note, etc. Some of the characteristics are shared by other books, like authors and subjects. In a relational database these shared characteristics would be saved in additional tables, and they would be joined to a specific book through the use of unique identifiers (keys). Given this sort of data structure, reports can be created from the database describing its content. Similarly, queries can be applied against the database to uncover relationships that may not be apparent at first glance or buried in reports. The power of relational databases lay in the use of keys to make relationships between rows in one table and rows in other tables.

Not coincidently, this is very much the way linked data is expected to be implemented. In the linked data world, the subjects of triples are URIs (think database keys). Each URI is associated with one or more predicates (think the characteristics in the book example). Each triple then has an object, and these objects take the form of literals or other URIs. In the book example, the object could be “Adventures Of Huckleberry Finn” or a URI pointing to Mark Twain. The reports of relational databases are analogous to RDF serializations, and SQL (the relational database query language) is analogous to SPARQL, the query language of RDF triple stores. Because of the close similarity between well-designed relational databases and linked data principles, the publishing of linked data directly from relational databases makes whole lot of sense, but the

process requires the combined time and skills of a number of different people: content specialists, database designers, and computer programmers. Consequently, the process of publishing linked data from relational databases may be optimal, but it is more expensive.

Thankfully, most archivists probably use some sort of database to manage their collections and create their finding aids. Moreover, archivists probably use one of three or four tools for this purpose: Archivist's Toolkit, Archon, ArchivesSpace, or PastPerfect. Each of these systems have a relational database at their heart. Reports could be written against the underlying databases to generate serialized RDF and thus begin the process of publishing linked data. Doing this from scratch would be difficult, as well as inefficient because many people would be starting out with the same database structure but creating a multitude of varying outputs. Consequently, there are two alternatives. The first is to use a generic database application to RDF publishing platform called D2RQ. The second is for the community to join together and create a holistic RDF publishing system based on the database(s) used in archives.

D2RQ is a wonderful software system. It is supported, well-documented, executable on just about any computing platform, open source, focused, functional, and at the same time does not try to be all things to all people. Using D2RQ it is more than possible to quickly and easily publish a well-designed relational database as RDF. The process is relatively simple:

1. download the software
   |
2. use a command-line utility to map the database structure to a configuration file

3. season the configuration file to taste

4. run the D2RQ server using the configuration file as input thus allowing people or RDF user-agents to search and browse the database using linked data principles

5. alternatively, dump the contents of the database to an RDF serialization and upload the result into your favorite RDF triple store

The downside of D2RQ is its generic nature. It will create an RDF ontology whose terms correspond to the names of database fields. These field names do not map to widely accepted ontologies and therefore will not interact well with communities outside the ones using a specific database structure. Still, the use of D2RQ is quick, easy, and accurate.

The second alternative to using databases of archival content to published linked data requires community effort and coordination. The databases of Archivist's Toolkit, Archon, ArchivesSpace, or Past Perfect could be assumed. The community could then get together and create and decide on an RDF ontology to use for archival descriptions. The database structure(s) could then be mapped to this ontology. Next, programs could be written against the database(s) to create serialized RDF thus beginning the process of publishing linked data. Once that was complete, the archival community would need to come together again to ensure it uses as many shared URIs as possible thus creating the most functional sets of linked data. This second alternative requires a significant amount of community involvement and wide-spread education. It represents a never-ending process.

On Your Way: Next Steps

Integration into daily practice

Three Cs: Cleanup, Conversion, Consistency

Creating and maintaining metadata is a never-ending process. The items being described can always use elaboration. Collections may increase is size. Rights applied against content may change. Things become digitized, or digitized things are migrated from one format to another. Because of these sorts of things and many others, cleanup, conversion, and consistency are something every metadata specialist needs to keep in mind.

Cleanup, conversion, and consistency means many things. Does all of your metadata use the same set of one or more vocabularies? Are things spelled correctly? Maybe you used abbreviations in one document but spelled things out in another? Have you migrated your JPEG images to JPEG2000 or TIFF formats? Maybe the EAD DTD has been updated, and you want (need) to migrate your finding aids from one XML format to another? Do all of your finding aids exhibit the same level of detail; are some "thinner" than others? Have you used one form of a person's name in one document but used another form in a different document? The answers to these sorts of questions point to the need for cleanup, conversion, and consistency.

Tools

- Fusion Tables (http://www.google.com/drive/apps.html) - Bust your data out of its silo! Combine it with other data on the web. Collaborate, visualize and share.

- OpenRefine (https://github.com/OpenRefine/) - OpenRefine is a free, open source power tool for working with messy data and improving it

- cURL - curl -L -H 'Accept: application/rdf+xml' http://infomotions.com/sandbox/liam/id/ctumarc15567

Looking Ahead: Advanced Tools and Visualizations

Tools for archivists (data preparation, cleanup, management)

What's available now

- Bibframe (http://bibframe.org) - The Bibliographic Framework Initiative (BIBFRAME) is an undertaking by the Library of Congress and the community to better accommodate future needs of the library community. A major focus of the initiative will be to determine a transition path for the MARC 21 exchange format to more Web based, Linked Data standards. Zepheira and The Library of Congress are working together to develop a Linked Data model, vocabulary and enabling tools / services for supporting this Initiative.

- ckan (http://ckan.org) - The open source data portal software

- CouchDB (http://couchdb.apache.org) - CouchDB is a database that completely embraces the web. Store your data with JSON documents. Access your documents with your web browser, via HTTP. Query, combine, and transform your documents with JavaScript. CouchDB works well with modern web and mobile apps. You can even serve web apps directly out of CouchDB. And you can distribute your data, or your apps, efficiently using CouchDB's incremental replication. CouchDB supports master-master setups with automatic conflict detection.

- Curl (http://curl.haxx.se) - curl is a command line tool for transferring data with URL syntax, supporting DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMTP, SMTPS, Telnet and TFTP. curl supports SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form based upload, proxies, cookies, user+password authentication (Basic, Digest, NTLM, Negotiate, kerberos...), file transfer resume, proxy tunneling and a busload of other useful tricks.

- D2RQ (http://d2rq.org) - The D2RQ Platform is a system for accessing relational databases as virtual, read-only RDF graphs. It offers RDF-based access to the content of relational databases without having to replicate it into an RDF store. Using D2RQ you can: query a non-RDF database using SPARQL, access the content of the database as Linked Data over the Web, create custom dumps of the database in RDF formats for loading into an RDF store, access information in a non-RDF database using the Apache Jena API

- Datahub (http://datahub.io/) - the free, powerful data management platform from the Open Knowledge Foundation

- Disco - Hyperdata Browser (http://wifo5-03.informatik.uni-mannheim.de/bizer/ng4j/disco/) - The Disco - Hyperdata Browser is a simple browser for navigating the Semantic Web as an unbound set of data sources. The browser renders all information, that it can find on the Semantic Web about a specific resource, as an HTML page. This resource description contains hyperlinks that allow you to navigate between resources. While you move from resource to resource, the browser dynamically retrieves information by dereferencing HTTP URIs and by following rdfs:seeAlso links.

- ead2rdf (http://data.archiveshub.ac.uk/xslt/ead2rdf.xsl) - The “transform” process is currently performed using XSLT to read an EAD XML document and output RDF/XML, and the current version of the stylesheet is now available:

- eaditor (https://github.com/ewg118/eaditor) - EADitor is an XForms framework for the creation and editing of Encoded Archival Description (EAD) finding aids using Orbeon, an enterprise-level XForms Java application, which runs in Apache Tomcat.

- Fusion Tables (http://www.google.com/drive/apps.html) - Bust your data out of its silo! Combine it with other data on the web. Collaborate, visualize and share.

- Linked Data Tools (http://linkeddata.org/tools) -

- Linked Media Framework (https://code.google.com/p/lmf/) - The Linked Media Framework is an easy-to-setup server application that bundles together some key open source projects to offer some advanced services for linked media management.

- oai2lod (https://github.com/behas/oai2lod) - exposes OAI-PMH data sources as Linked Data

- OpenLink Data Explorer Extension (http://ode.openlinksw.com) - The OpenLink Data Explorer (ODE) is a browser extension (currently available for Firefox, Safari, Chrome, Opera, and Internet Explorer with additional browser support to follow) that adds a new option to the realm of Web User Agent functionality, in the form of new menu options for viewing Data Sources associated with Web Pages.

- OpenRefine (https://github.com/OpenRefine/) - OpenRefine is a free, open source power tool for working with messy data and improving it

- Perl and RDF (http://www.perlrdf.org) - The Perl RDF project hopes to address these issues:, publish an official API for storage, parsing and serializing modules, produce a set of base classes for representing common RDF objects such as statements and nodes (resources, literals, blank nodes), produce patches to existing RDF tools to support these APIs, subclassing where appropriate, produce a test suite for storage, parsing, serializing, statement and node classes.

- Perl-SPARQL-client-library (https://github.com/swh/Perl-SPARQL-client-library) - A simple Perl library for accessing SPARQL endpoints.

- Protégé (http://protege.stanford.edu) - Protégé is a free, open source ontology editor and knowledge-base framework The Protégé platform supports modeling ontologies via a web client or a desktop client. Protégé ontologies can be developed in a variety of formats including OWL, RDF(S), and XML Schema Protégé is based on Java, is

extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development.

- RDFImportersAndAdapters (http://www.w3.org/wiki/RDFImportersAndAdapters) - Tools and applications that can convert from other data and file formats to RDF.

- Semantic Web Development Tools (http://www.w3.org/2001/sw/wiki/Tools) - This Wiki contains a collection of tool references that can help in developing Semantic Web applications. These include complete development environments, editors, libraries or modules for various programming languages, specialized browsers, etc. The goal is to list such tools and not Semantic Web applications in general (the interested reader may consider looking at the W3C SW Use Case Collection for those.)

- Sematic Web Client Library (http://wifo5-03.informatik.uni-mannheim.de/bizer/ng4j/semwebclient/) - The Sematic Web Client Library represents the complete Semantic Web as a single RDF graph. The library enables applications to query this global graph using SPARQL- and find(SPO) queries. To answer queries, the library dynamically retrieves information from the Semantic Web by dereferencing HTTP URIs, by following rdfs:seeAlso links, and by querying the Sindice search engine. The library is written in Java and is based on the Jena framework.

- SparqlImplementations (http://www.w3.org/wiki/SparqlImplementations) - This page lists some implementations of SPARQL, a query language and protocol for RDF acccess released by the W3C RDF Data Access Working Group - DAWG.

- Tableau Public (http://www.tableausoftware.com/public) - With Tableau Public you can create interactive graphs, dashboards, maps and tables from virtually any data and embed them on your website or blog in minutes.

- Tabulator (http://www.w3.org/2005/ajar/tab) - The Tabulator project is a generic data browser and editor. Using outline and table modes, it provides a way to browse RDF data on the web. RDF is the standard for inter-application data exchange.

- TemaTres (http://www.vocabularyserver.com) - The open source way to manage formal representations of knowledge

- VirtuosoUniversalServer (http://www.w3.org/wiki/VirtuosoUniversalServer) - OpenLink Virtuoso is a multi-purpose and multi-protocol (Hybrid) Data Server from OpenLink Software that includes SQL Object-Relational, RDF, XML, and Free Text data management, alongside Web Application (HTTP, SOAP, WebDAV), SyncML, and Discussion Server functionality, in a single server.

- W3C RDF Validation Service (http://www.w3.org/RDF/Validator/) - Enter a URI or paste an RDF/XML document into the text field above. A 3-tuple (triple) representation of the corresponding data model as well as an optional graphical visualization of the data model will be displayed.

Gaps: What is needed

There needs to be easy to use tools to find URIs and insert them in to archival descriptions. One such tool is called lobid:

> In “From strings to things: A linked data API for library hackers and Web developers” Fabian Steeg and Pascal Christoph (HBZ) described an interface allowing librarians to determine the URIs of people, places, and things for library catalog records. “How can we benefit from linked data without being linked data experts? We want to pub Web developers into focus using JSON for HTTP.” There are few hacks illustrating some of their work on Github in the lobid repository. --https://github.com/lobid

Another example would be an interface to the varius linked data sets available from the Library of Congress. --http://id.loc.gov

Tools for users: visualizations, interfaces, etc.

What's available now

- D3.js (http://d3js.org) - D3.js is a JavaScript library for manipulating documents based on data. D3 helps you bring data to life using HTML, SVG and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

- Gephi (http://gephi.org) - Gephi is an interactive visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs.

- Tableau Public (http://www.tableausoftware.com/public) - With Tableau Public you can create interactive graphs, dashboards, maps and tables from virtually any data and embed them on your website or blog in minutes.

Gaps: What is needed

Tools

- “4store - Scalable RDF Storage.” Accessed November 12, 2013. http://4store.org/.
- “Apache Jena - Home.” Accessed November 11, 2013. http://jena.apache.org/.
- “Behas/oai2lod · GitHub.” Accessed November 3, 2013. https://github.com/behas/oai2lod.
- “BIBFRAME.ORG :: Bibliographic Framework Initiative - Overview.” Accessed November 3, 2013. http://bibframe.org/.
- “Ckan - The Open Source Data Portal Software.” Accessed November 3, 2013. http://ckan.org/.
- “Community | Tableau Public.” Accessed November 3, 2013. http://www.tableausoftware.com/public/community.
- “ConverterToRdf - W3C Wiki.” Accessed November 11, 2013. http://www.w3.org/wiki/ConverterToRdf.
- “Curl and Libcurl.” Accessed November 3, 2013. http://curl.haxx.se/.
- “D2R Server | The D2RQ Platform.” Accessed November 15, 2013. http://d2rq.org/d2r-server.
- “Disco Hyperdata Browser.” Accessed November 3, 2013. http://wifo5-03.informatik.uni-mannheim.de/bizer/ng4j/disco/.
- “Ead2rdf.” Accessed November 3, 2013. http://data.archiveshub.ac.uk/xslt/ead2rdf.xsl.

- “Ewg118/eaditor · GitHub.” Accessed November 3, 2013. https://github.com/ewg118/eaditor.
- “Google Drive.” Accessed November 3, 2013. http://www.google.com/drive/apps.html.
- Library, The standard EAC-CPF is maintained by the Society of American Archivists in partnership with the Berlin State. “Society of American Archivists and the Berlin State Library.” Accessed January 1, 2014. http://eac.staatsbibliothek-berlin.de/.
- “Lmf - Linked Media Framework - Google Project Hosting.” Accessed November 3, 2013. https://code.google.com/p/lmf/.
- “OpenLink Data Explorer Extension.” Accessed November 3, 2013. http://ode.openlinksw.com/.
- “openRDF.org: Home.” Accessed November 12, 2013. http://www.openrdf.org/.
- “OpenRefine (OpenRefine) · GitHub.” Accessed November 3, 2013. https://github.com/OpenRefine/.
- “Parrot, a RIF and OWL Documentation Service.” Accessed November 11, 2013. http://ontorule-project.eu/parrot/parrot.
- “RDF2RDF - Converts RDF from Any Format to Any.” Accessed December 5, 2013. http://www.l3s.de/~minack/rdf2rdf/.
- “RDFImportersAndAdapters - W3C Wiki.” Accessed November 3, 2013. http://www.w3.org/wiki/RDFImportersAndAdapters.
- “RDFizers - SIMILE.” Accessed November 11, 2013. http://simile.mit.edu/wiki/RDFizers.

- “Semantic Web Client Library.” Accessed November 3, 2013. http://wifo5-03.informatik.uni-mannheim.de/bizer/ng4j/semwebclient/.
- “SIMILE Widgets | Exhibit.” Accessed November 11, 2013. http://www.simile-widgets.org/exhibit/.
- “SparqlImplementations - W3C Wiki.” Accessed November 3, 2013. http://www.w3.org/wiki/SparqlImplementations.
- “swh/Perl-SPARQL-client-library · GitHub.” Accessed November 3, 2013. https://github.com/swh/Perl-SPARQL-client-library.
- “Tabulator: Generic Data Browser.” Accessed November 3, 2013. http://www.w3.org/2005/ajar/tab.
- “TaskForces/CommunityProjects/LinkingOpenData/SemWebClients - W3C Wiki.” Accessed November 5, 2013. http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/SemWebClients.
- “TemaTres Controlled Vocabulary Server.” Accessed November 3, 2013. http://www.vocabularyserver.com/.
- “The D2RQ Platform – Accessing Relational Databases as Virtual RDF Graphs.” Accessed November 3, 2013. http://d2rq.org/.
- “The Protégé Ontology Editor and Knowledge Acquisition System.” Accessed November 3, 2013. http://protege.stanford.edu/.
- “Tools - Semantic Web Standards.” Accessed November 3, 2013. http://www.w3.org/2001/sw/wiki/Tools.
- “Tools | Linked Data - Connect Distributed Data Across the Web.” Accessed November 3, 2013. http://linkeddata.org/tools.

- “Vapour, a Linked Data Validator.” Accessed November 11, 2013. http://validator.linkeddata.org/vapour.

- “VirtuosoUniversalServer - W3C Wiki.” Accessed November 3, 2013. http://www.w3.org/wiki/VirtuosoUniversalServer.

- “W3C RDF Validation Service.” Accessed November 3, 2013. http://www.w3.org/RDF/Validator/.

- “W3c/rdfvalidator-ng.” Accessed December 10, 2013. https://github.com/w3c/rdfvalidator-ng.

- “Working with RDF with Perl.” Accessed November 3, 2013. http://www.perlrdf.org/.

Data sets

- “(LOV) Linked Open Vocabularies.” Accessed November 3, 2013. http://lov.okfn.org/dataset/lov/.
- “Data Sets & Services.” Accessed November 3, 2013. http://www.oclc.org/data/data-sets-services.en.html.
- “Data.gov.uk.” Accessed November 3, 2013. http://data.gov.uk/.
- “Freebase.” Accessed November 3, 2013. http://www.freebase.com/.
- “GeoKnow/LinkedGeoData · GitHub.” Accessed November 3, 2013. https://github.com/GeoKnow/LinkedGeoData.
- “GeoNames.” Accessed November 3, 2013. http://www.geonames.org/.
- “Getty Union List of Artist Names (Research at the Getty).” Accessed November 3, 2013. http://www.getty.edu/research/tools/vocabularies/ulan/.
- “Home - LC Linked Data Service (Library of Congress).” Accessed November 3, 2013. http://id.loc.gov/.
- “Home | Data.gov.” Accessed November 3, 2013. http://www.data.gov/.
- “ISBNdb - A Unique Book & ISBN Database.” Accessed November 3, 2013. http://isbndb.com/.
- “Linked Movie Data Base | Start Page.” Accessed November 3, 2013. http://linkedmdb.org/.
- “MusicBrainz - The Open Music Encyclopedia.” Accessed November 3, 2013. http://musicbrainz.org/.

- “New York Times - Linked Open Data.” Accessed November 3, 2013. http://data.nytimes.com/.
- “PELAGIOS: About PELAGIOS.” Accessed September 4, 2013. http://pelagios-project.blogspot.com/p/about-pelagios.html.
- “Start Page | D2R Server for the CIA Factbook.” Accessed November 3, 2013. http://wifo5-03.informatik.uni-mannheim.de/factbook/.
- “Start Page | D2R Server for the Gutenberg Project.” Accessed November 3, 2013. http://wifo5-03.informatik.uni-mannheim.de/gutendata/.
- “The Friend of a Friend (FOAF) Project | FOAF Project.” Accessed November 3, 2013. http://www.foaf-project.org/.
- “VIAF.” Accessed August 27, 2013. http://viaf.org/.
- “VoID - Semanticweb.org.” Accessed November 3, 2013. http://semanticweb.org/wiki/VoID.
- “Web Data Commons.” Accessed November 19, 2013. http://webdatacommons.org/.
- “Welcome - the Datahub.” Accessed August 14, 2013. http://datahub.io/.
- “Welcome to Open Library (Open Library).” Accessed November 3, 2013. https://openlibrary.org/.
- “Wiki.dbpedia.org : About.” Accessed November 3, 2013. http://dbpedia.org/About.
- “World Bank Linked Data.” Accessed November 3, 2013. http://worldbank.270a.info/.html.

Further reading

This is a list of links and citations to get one started on Linked Open Data

- admin. “Barriers to Using EAD,” August 4, 2012. http://oclc.org/research/activities/eadtools.html.

- Becker, Christian, and Christian Bizer. “Exploring the Geospatial Semantic Web with DBpedia Mobile.” Web Semantics: Science, Services and Agents on the World Wide Web 7, no. 4 (December 2009): 278–286. doi:10.1016/j.websem.2009.09.004.

- Belleau, François, Marc-Alexandre Nolin, Nicole Tourigny, Philippe Rigault, and Jean Morissette. “Bio2RDF: Towards a Mashup to Build Bioinformatics Knowledge Systems.” Journal of Biomedical Informatics 41, no. 5 (October 2008): 706–716. doi:10.1016/j.jbi.2008.03.004.

- Berners-Lee, Tim. “Linked Data - Design Issues.” Accessed August 4, 2013. http://www.w3.org/DesignIssues/LinkedData.html.

- Berners-Lee, Tim, James Hendler, and Ora Lassila. “The Semantic Web.” Scientific American 284, no. 5 (May 2001): 34–43. doi:10.1038/scientificamerican0501-34.

- Bizer, Christian, Tom Heath, and Tim Berners-Lee. “Linked Data - The Story So Far:” International Journal on Semantic Web and Information Systems 5, no. 3 (33 2009): 1–22. doi:10.4018/jswis.2009081901.

- Carroll, Jeremy J., Christian Bizer, Pat Hayes, and Patrick Stickler. “Named Graphs.” Web Semantics: Science, Services and Agents on the World Wide Web 3, no. 4 (December 2005): 247–267. doi:10.1016/j.websem.2005.09.001.

- “Chem2bio2rdf - How to Publish Data Using D2R?” Accessed January 6, 2014. http://chem2bio2rdf.wikispaces.com/How+to+publish+data

+using+D2R%3F.

- “Content Negotiation.” Wikipedia, the Free Encyclopedia, July 2, 2013. https://en.wikipedia.org/wiki/Content_negotiation.

- “Cool URIs for the Semantic Web.” Accessed November 3, 2013. http://www.w3.org/TR/cooluris/.

- Correndo, Gianluca, Manuel Salvadores, Ian Millard, Hugh Glaser, and Nigel Shadbolt. “SPARQL Query Rewriting for Implementing Data Integration over Linked Data.” 1. ACM Press, 2010. doi: 10.1145/1754239.1754244.

- David Beckett. “Turtle.” Accessed August 6, 2013. http://www.w3.org/TR/2012/WD-turtle-20120710/.

- “Debugging Semantic Web Sites with cURL | Cygri’s Notes on Web Data.” Accessed November 3, 2013. http://richard.cyganiak.de/blog/2007/02/debugging-semantic-web-sites-with-curl/.

- Dunsire, Gordon, Corey Harper, Diane Hillmann, and Jon Phipps. “Linked Data Vocabulary Management: Infrastructure Support, Data Integration, and Interoperability.” Information Standards Quarterly 24, no. 2/3 (2012): 4. doi:10.3789/isqv24n2-3.2012.02.

- Elliott, Thomas, Sebastian Heath, and John Muccigrosso. “Report on the Linked Ancient World Data Institute.” Information Standards Quarterly 24, no. 2/3 (2012): 43. doi:10.3789/isqv24n2-3.2012.08.

- Fons, Ted, Jeff Penka, and Richard Wallis. “OCLC’s Linked Data Initiative: Using Schema.org to Make Library Data Relevant on the Web.” Information Standards Quarterly 24, no. 2/3 (2012): 29. doi: 10.3789/isqv24n2-3.2012.05.

- Hartig, Olaf. “Querying Trust in RDF Data with tSPARQL.” In The Semantic Web: Research and Applications, edited by Lora Aroyo, Paolo

Traverso, Fabio Ciravegna, Philipp Cimiano, Tom Heath, Eero Hyvönen, Riichiro Mizoguchi, Eyal Oren, Marta Sabou, and Elena Simperl, 5554:5–20. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. http://www.springerlink.com/index/10.1007/978-3-642-02121-3_5.

- Hartig, Olaf, Christian Bizer, and Johann-Christoph Freytag. "Executing SPARQL Queries over the Web of Linked Data." In The Semantic Web - ISWC 2009, edited by Abraham Bernstein, David R. Karger, Tom Heath, Lee Feigenbaum, Diana Maynard, Enrico Motta, and Krishnaprasad Thirunarayan, 5823:293–309. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. http://www.springerlink.com/index/10.1007/978-3-642-04930-9_19.

- Heath, Tom, and Christian Bizer. "Linked Data: Evolving the Web into a Global Data Space." Synthesis Lectures on the Semantic Web: Theory and Technology 1, no. 1 (February 9, 2011): 1–136. doi:10.2200/S00334ED1V01Y201102WBE001.

- Isaac, Antoine, Robina Clayphan, and Bernhard Haslhofer. "Europeana: Moving to Linked Open Data." Information Standards Quarterly 24, no. 2/3 (2012): 34. doi:10.3789/isqv24n2-3.2012.06.

- Kobilarov, Georgi, Tom Scott, Yves Raimond, Silver Oliver, Chris Sizemore, Michael Smethurst, Christian Bizer, and Robert Lee. "Media Meets Semantic Web – How the BBC Uses DBpedia and Linked Data to Make Connections." In The Semantic Web: Research and Applications, edited by Lora Aroyo, Paolo Traverso, Fabio Ciravegna, Philipp Cimiano, Tom Heath, Eero Hyvönen, Riichiro Mizoguchi, Eyal Oren, Marta Sabou, and Elena Simperl, 5554:723–737. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. http://www.springerlink.com/index/10.1007/978-3-642-02121-3_53.

- LiAM. "LiAM: Linked Archival Metadata." Accessed July 30, 2013. http://sites.tufts.edu/liam/.

- “Linked Data.” Wikipedia, the Free Encyclopedia, July 13, 2013. http://en.wikipedia.org/w/index.php?title=Linked_data&oldid=562554554.

- “Linked Data Glossary.” Accessed January 1, 2014. http://www.w3.org/TR/ld-glossary/.

- “Linked Open Data.” Europeana. Accessed September 12, 2013. http://pro.europeana.eu/web/guest;jsessionid=09A5D79E7474609AE246DF5C5A18DDD4.

- “Linked Open Data in Libraries, Archives, & Museums (Google Group).” Accessed August 6, 2013. https://groups.google.com/forum/#!forum/lod-lam.

- “Linking Lives | Using Linked Data to Create Biographical Resources.” Accessed August 16, 2013. http://archiveshub.ac.uk/linkinglives/.
- “LOCAH Linked Archives Hub Test Dataset.” Accessed August 6, 2013. http://data.archiveshub.ac.uk/.

- “LODLAM - Linked Open Data in Libraries, Archives & Museums.” Accessed August 6, 2013. http://lodlam.net/.

- “Notation3.” Wikipedia, the Free Encyclopedia, July 13, 2013. http://en.wikipedia.org/w/index.php?title=Notation3&oldid=541302540.

- “OWL 2 Web Ontology Language Primer.” Accessed August 14, 2013. http://www.w3.org/TR/2009/REC-owl2-primer-20091027/.

- Quilitz, Bastian, and Ulf Leser. “Querying Distributed RDF Data Sources with SPARQL.” In The Semantic Web: Research and Applications, edited by Sean Bechhofer, Manfred Hauswirth, Jörg Hoffmann, and Manolis Koubarakis, 5021:524–538. Berlin, Heidelberg: Springer Berlin Heidelberg. Accessed September 4, 2013. http://www.springerlink.com/index/10.1007/978-3-540-68234-9_39.

- “RDF/XML.” Wikipedia, the Free Encyclopedia, July 13, 2013. http://en.wikipedia.org/wiki/RDF/XML.

- “RDFa.” Wikipedia, the Free Encyclopedia, July 22, 2013. http://en.wikipedia.org/wiki/RDFa.

- “Semantic Web.” Wikipedia, the Free Encyclopedia, August 2, 2013. http://en.wikipedia.org/w/index.php?title=Semantic_Web&oldid=566813312.

- “SPARQL.” Wikipedia, the Free Encyclopedia, August 1, 2013. http://en.wikipedia.org/w/index.php?title=SPARQL&oldid=566718788.
- “SPARQL 1.1 Overview.” Accessed August 6, 2013. http://www.w3.org/TR/sparql11-overview/.

- “Spring/Summer 2012 (v.24 No.2/3) - National Information Standards Organization.” Accessed August 6, 2013. http://www.niso.org/publications/isq/2012/v24no2-3/.

- Summers, Ed, and Dorothea Salo. Linking Things on the Web: A Pragmatic Examination of Linked Data for Libraries, Archives and Museums. ArXiv e-print, February 19, 2013. http://arxiv.org/abs/1302.4591.

- “The Linking Open Data Cloud Diagram.” Accessed November 3, 2013. http://lod-cloud.net/.

- “The Trouble with Triples | Duke Collaboratory for Classics Computing (DC3).” Accessed November 6, 2013. http://blogs.library.duke.edu/dcthree/2013/07/27/the-trouble-with-triples/.

- Tim Berners-Lee, James Hendler, and Ora Lassila. “The Semantic Web.” Accessed September 4, 2013. http://www.scientificamerican.com/article.cfm?id=the-semantic-web.

- “Transforming EAD XML into RDF/XML Using XSLT.” Accessed August 16, 2013. http://archiveshub.ac.uk/locah/tag/transform/.

- “Triplestore - Wikipedia, the Free Encyclopedia.” Accessed November 11, 2013. http://en.wikipedia.org/wiki/Triplestore.

- “Turtle (syntax).” Wikipedia, the Free Encyclopedia, July 13, 2013. http://en.wikipedia.org/w/index.php?title=Turtle_(syntax)&oldid=542183836.

- Volz, Julius, Christian Bizer, Martin Gaedke, and Georgi Kobilarov. “Discovering and Maintaining Links on the Web of Data.” In The Semantic Web - ISWC 2009, edited by Abraham Bernstein, David R. Karger, Tom Heath, Lee Feigenbaum, Diana Maynard, Enrico Motta, and Krishnaprasad Thirunarayan, 5823:650–665. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. http://www.springerlink.com/index/10.1007/978-3-642-04930-9_41.

- Voss, Jon. “LODLAM State of Affairs.” Information Standards Quarterly 24, no. 2/3 (2012): 41. doi:10.3789/isqv24n2-3.2012.07.

- W3C. “LinkedData.” Accessed August 4, 2013. http://www.w3.org/wiki/LinkedData.

- “Welcome to Euclid.” Accessed September 4, 2013. http://www.euclid-project.eu/.

- “Wiki.dbpedia.org : About.” Accessed November 3, 2013. http://dbpedia.org/About.