Chapter 2

Semantic Web and Linked Data

Abstract

Chapter 2 introduces Semantic Web concepts in the context of library data and defines common Semantic Web terminology and acronyms, as well as the primary components of linked data: open data, machine-readable data formats, and the use of Uniform Resource Identifiers (URIs).

The World Wide Web was invented because Tim Berners-Lee, a scientist at the CERN nuclear research laboratory in Switzerland, wanted a way for him and his colleagues to share documents over the Internet. The genius of the invention was the ability to link from one document to another, thus creating a digital and actionable version of the classic citation. The Semantic Web is also about linking, but it adds to the original Web the linking of data, not just documents. It also changes the nature of the link: whereas the link between documents has no meaning other than "link," in the Semantic Web the links themselves have a specific meaning. We can illustrate this using the citation example: in a standard document, a citation is simply a number in the text and a bibliographic citation at the end of the page. You don't know why the author is citing that work other than what you can glean from the surrounding text. Using the richly semantic links of the Semantic Web, you could characterize each citation with a meaning such as "cites as evidence," "disagrees with," or "extends." (Those examples are from an actual Semantic Web vocabulary, CiTO, which will be described later in this report.)

There are two ways that the Semantic Web will be built: by linking information that exists within documents, and by allowing data itself to be on the Web. Markup of information in documents could allow smarter access to that information than we get with keyword searching. For example, markup could identify the author of a document so that an actual author search could be done, something that our search engines do not provide today. It could also add machine actionability to information in texts. While you and I can easily interpret "Herman Melville, the author of *Moby-Dick*," an indexing algorithm sees that as merely a string of seven units that can be indexed. Adding markup that specifies that the string "Herman Melville" represents a person, that "*Moby-Dick*" represents a book, and that there is a relationship called "author" makes it possible to query the Web with "Who is the author of *Moby-Dick*?" or "What book(s) did Herman Melville write?" This type of markup is the goal of microformats, although not all microformats use Semantic Web standards. (Microformats are discussed below.)

The second method of populating the Semantic Web is that of adding actual datasets to the Web. This is the growing area that is visible on the linked data cloud. This is also the area that is of most interest at the moment to libraries because the library catalog itself qualifies as data that can become part of the linked data cloud.

> *Linked Data*
> http://linkeddata.org

While the emphasis in this report is on libraries and linked data, it is useful to note here that much of the work on Semantic Web technologies is taking place in the scientific arena, in particular in biomedical research, and in the realm of government data. The World Wide Web Consortium (W3C) has active interest groups in these two areas. This does not mean that

library data is not on the Semantic Web agenda. In fact, the W3C sponsored a Library Linked Data Incubator Group in 2010–2011 to investigate possibilities for library data on the Web.[1] Libraries of course provide the foundation for knowledge-creation activities, and having library data available in this growing Web of data will help make the connection between new knowledge and the research that it cites.

Semantic Web Basics

This section will cover some common Semantic Web terminology and acronyms that you will encounter in documents and discussions. This will not be enough detail to allow the reader to become a bona fide Semantic Web developer. If that is your goal, you should spend time on the Semantic Web pages of the W3C site. I also recommend some reading in chapter 6.

RDF and the Triple

The Resource Description Framework (RDF) is a formal language that defines the basic structure of the linked data that makes up the Semantic Web. If I can stretch an analogy a bit, RDF is to the Semantic Web as data packets are to the Internet. Both provide a basic, underlying structure that services can be built upon. They both are designed for use by computer programs, not by humans. But here's the rub: there are very few user-friendly applications and services in existence today that make use of RDF. Developmentally the Semantic Web is where the WWW was before Marc Andreessen and colleagues developed the first Web browser, called Mosaic, and released it publicly in 1993. This means that users of linked data today are generally programmers and developers who are comfortable working directly with what is under the hood of this new technology. The rest of us are impatiently waiting for the user-friendly interface that will let us easily make use of linked data.

Simply put, RDF defines the basic unit of the Semantic Web as a three-part structure, commonly referred to as a triple. This structure is analogous to a very simple sentence, and each triple has this same set of components:

subject → predicate → object

The subject is what you are talking about, the object is what you are saying about it, and the predicate is a verb-like connector that states meaningfully what links the subject and object. For example, if you are describing a book and its author, you could create a statement like this:

Moby-Dick → has author → Herman Melville

While the structure is called a triple, a triple of information is often referred to as a statement because it states some information about the subject.

When someone says that data has been made available "in RDF," that is usually shorthand for saying that the data follows Semantic Web standards. You will often see references to RDF/XML. Data in that format uses the standard RDF schema, also shortened to "RDFs."

SKOS

The Simple Knowledge Organization System (SKOS) is one of the first structures built on top of RDF, and it is proving to be very useful. SKOS is a standard for encoding thesauri and controlled lists. It includes the basic structure of a thesaurus, including the concepts of broader, narrower, and related that can be applied between entries. It also allows for the designation of what librarians would call the "authoritative" display term and what SKOS calls a "preferred label" (shortened to `prefLabel` in encoded SKOS). There are also alternate labels and hidden labels allowed in SKOS, which can provide a variety of entry points for searching. Because SKOS is a Web standard and the Web is global and multilingual, any of these display labels can be encoded by language, and the labels for any SKOS entry can be provided in as many languages as desired. There are some SKOS examples at the beginning of chapter 4.

OWL

Briefly, the Web Ontology Language (OWL) is a standard that extends RDF and is used to define specific Semantic Web metadata vocabularies (also called ontologies). For example, if you wish to express your warehouse data as linked data, you would use OWL to explain in machine language what your metadata is and how it relates to other data in the Web of data. OWL is to be used by the developers of metadata formats for the Semantic Web; as such, it is quite complex. OWL has already been through its own development cycle and as a result exists in a small number of versions. Most of us will never work directly with OWL, but if a metadata standard is "in OWL," you can know that it is designed to be Semantic Web–compliant.

SPARQL

One vision of the Semantic Web is that it is a huge web of data that uses the WWW as its database platform. In fact, it is expected that the Semantic Web will be queried much as a database is queried. A standard query language for that purpose, SPARQL (pronounced "sparkle"), is designed specifically to query the underlying triples of the Semantic Web using an SQL-like query format. Because SPARQL is designed to

be run against Web resources, you must first point the SPARQL engine at a dataset. You can query on one, two, or all three elements of the triple. Here's a sample SPARQL query from the tutorial SPARQL by Example (see chapter 6).

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?craft ?homepage
{
   ?craft foaf:name "Apollo 7" .
   ?craft foaf:homepage ?homepage
}
```

RDFa and Microformats

Markup within HTML tagging is called a *microformat*—a kind of format within a format. Microformats are designed for automated processing of data within webpages, which might be natural language or formatted displays of product information. In either case, eye-readable displays are not machine actionable, and the microformats embed tagged information for automated processing.

With the intention of fulfilling the vision of using semantic markup within documents, the W3C developed RDFa, a set of XHTML attributes that can be used to define data and links within a Web document. Because of the complexity of RDF, however, not everyone considers RDFa a viable standard for markup of webpages. The major search engines announced a lighter-weight format called Schema.org. This markup language is intended specifically for search rather than for the linking that is the hallmark of the Semantic Web. In response, the Semantic Web community has developed Schema.RDFS.org, an RDF-compliant version of the metadata defined by Schema.org that is updated to match the Schema.org metadata as it evolves. These microforms will be covered in more detail in chapter 3.

Schema.org
http://schema.org

Schema.RDFS.org
http://schema.rdfs.org

Linked Data: Four Rules, Five Stars, and a Plan

Linked data is not a single standard or format but, as Tim Berners-Lee says in the informal document that first stated the four rules of linked data, it is an "expectation of behavior." In design terms, Berners-Lee defined that behavior in this way:

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

The emphasis here is on the use of identifiers, and in fact that is a key element of linked data. RDF is designed for machine "understanding" and therefore does not use natural language in its statements. In fact, the subject and predicate must be identifiers, and the identifiers are in the form of a URL (called a URI because it identifies, not locates, as its function). These rules simply say that you should use identifiers for the things you are describing with your metadata and for your metadata itself. Those identifiers will be more precise than natural language, they will be language-neutral, and because the identifier takes the form "http://" it can also be used to provide information at that location about the thing it identifies.

The five stars, often seen on the coffee cup design in figure 2.1, define linked data that adheres to the Semantic Web standards. This is a high-level view, and it needs some filling in before it can become a plan. One possible plan is laid out in the Singapore Framework for Dublin Core Application Profiles developed by the Dublin Core community. As described in the step-by-step document "Guidelines for Dublin Core Application Profiles," the steps are

5. **Define your model.** This is your definition of the things your metadata will describe, such as *documents* and *persons*, and the relationships between them. FRBR is this kind of model.
6. **Select (or define) your metadata terms.** These are the data elements that your metadata will use. Note that this step emphasizes selecting, where possible, terms that have already been defined; new terms should be created only if none exists. The terms you use need to be defined using an RDF-based standard (often either RDF itself or OWL). Following those standards, each term will be identified with a URI, and the description of the term and any information about it (scope notes, etc.) should be openly available on the Web. The term will always be referred to using the URI, which is its identity in the Web of linked data. A human-friendly display of the term (or multiple displays in different languages) is always available, and ideally the URI points to all of the relevant information about the term, using Semantic Web standards, as well as helpful notes on its meaning and use.
7. **Select or define any controlled vocabularies you will use.** Ideally, each list of terms will be described in an RDF-compatible format (often

SKOS), and each term in such a vocabulary will be identified with a URI. This URI then allows for a flexible display of the vocabulary using different languages.

8. **Create links from your data to related data on the Web.** This is not truly the last step because some linking occurs throughout the process of developing your data elements. When you use an RDF-defined metadata term or vocabulary that is used broadly, you have in fact created a link to every other use of that term. If you have defined new terms, however, you will need to create a relationship to similar (or broader or narrower) terms so that all of your metadata has hooks into the Web of linked data.[3]

This process is significantly different from the way that metadata was created in the unlinked world. In the past, when you planned new metadata, you had to take into account only your known data-sharing partners and develop a standard that all could agree upon. In the linked data world, the scope of sharing has become the entire World Wide Web. This opens up your data for greater use, but it also means that you need to think broadly about how your data fits with such a large information base. It helps to think about universals in your metadata, things like *people*, *places*, and *physical description*. These are concepts that are not limited to any one community but will be useful in many data contexts. These common elements are obvious points for linking and should not be seen as internal to any one community's data. At the same time, the first community to define useful terms is contributing them to the general pool of useful terms and elements that anyone can take advantage of. In this sense, libraries, with their extensive set of controlled vocabularies (including those in authority files), have a lot to contribute to shared linked data.

The Cloud

The very first point on the five-star linked data cup is, "On the web, open license." For your data to participate on the Web, it has to be openly accessible and usable. This does not mean that you could not create a closed linked data system for your own purposes, and in fact there is considerable attention at this time to the creation of enterprise systems using linked data. But we presume that libraries and other cultural heritage institutions will wish to contribute to the open exchange of information on the Web and the knowledge-creation activities that the Web of data will foster. This open data Web is visualized in the linked data cloud, and you will often hear the expression *linked open data* (LOD) referring to data that is available for unfettered use on the Web.

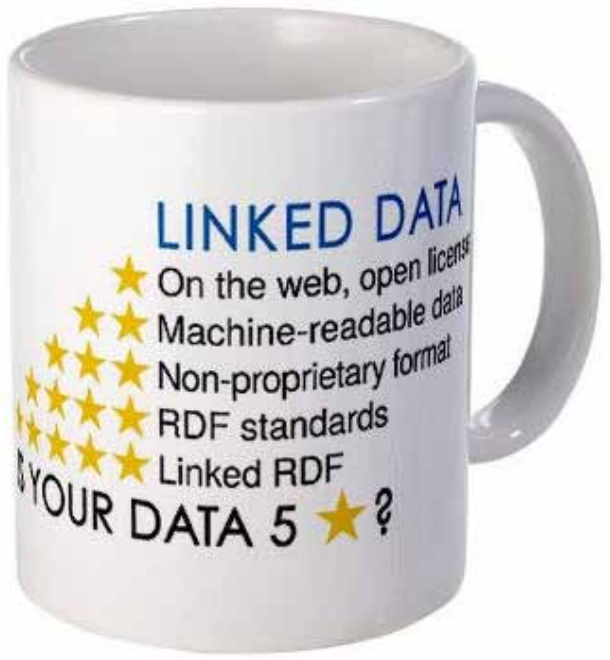

Figure 2.1
W3C's five-star data mug

It is worth taking a short look at the cloud itself as it exists today. We fortunately have a picture of the cloud that can help us visualize the datasets that are there and the connections between them. Beginning in 2007, when the cloud had only 12 datasets, Richard Cyganiak, of DERI in Ireland, and Anja Jentzsch, of the Freie Universität Berlin, have created a picture of the linked datasets and their connections. This graph now has over 300 members, and it is hard to take it in as a single picture.[4] These are not all of the sets of linked data in the world; the creators select only those with a significant number of links between them. The Web of data is growing by leaps and bound, not gradually, because many large datasets are being added from existing applications. There also is no comprehensive search engine for this data, so discovering previously unknown data of interest is still problematic. Those of you who remember the early days of the Internet with finding aids like Archie will understand the state of the art today.

As the cloud diagram has grown, it has become useful to gather the entries into categories based on the type of data and the community they serve. The categories and some examples from each are shown in table 2.1.

The center of the cloud is DBpedia (figure 2.2), a resource worth a few paragraphs on its own.

DBpedia is an extraction of data from the information boxes of *Wikipedia.* You will have seen information boxes in the upper right of each *Wikipedia* page, but there are many that appear throughout a

Categories	Examples
Media	BBC Programmes New York Times Music Brainz
Geographic	Ocean Drilling Codices Metoffice Weather Forecasts GeoNames
Publications	Manchest Reading Lists Sudoc Open Library LCSH
User-generated content	Flickr Semantic Tweet
Government	data.gove.uk Traffic Scotland Open Election Data Project
Cross-domain	Freebase Sears Linked Open Numbers
Life sciences	PubMed ChemBL GeneID

Table 2.1
Categories used in the LOD cloud diagram. Source: Anja Jentzsch, "LOD Cloud Diagram," Wikimedia Commons, September 2011, http://en.wikipedia.org/wiki/File:LOD_Cloud_Diagram_as_of_September_2011.png.

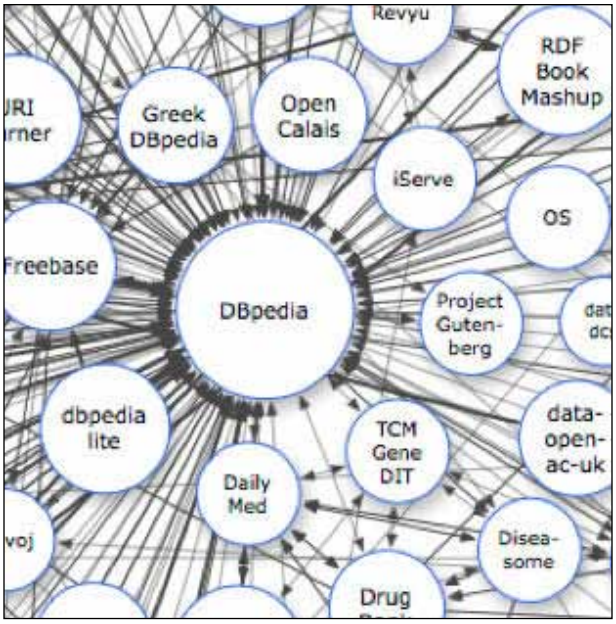

Figure 2.2
Center of the LOD cloud diagram

Wikipedia entry even though they may be less noticeable in the display. Each information box is a set of structured data, like dates, longitude and latitude, or the offices held by an elected official. The information boxes are specific to the type of data: those for people are different from those for places or events or technologies.

Anja Jentzsch, one of the creators of the linked data cloud diagram, has characterized DBpedia as "querying Wikipedia like a database."[5] DBpedia allows structured queries that are more information-rich than the simple keyword search within *Wikipedia* itself. Because *Wikipedia* is encyclopedic in the information it contains, DBpedia is as well. This makes DBpedia an ideal meeting point for a wide variety of linked data.

You can visit DBpedia and see a great deal of documentation about the service. You will not find what most of us would consider a user-friendly interface to the data, however. DBpedia is currently intended to serve as a large set of linking data for those who are developing Semantic Web applications. There is a searchable database called Freebase that has ingested much of the DBpedia data, along with data from other linked data sources. Freebase has a friendly interface and can give some idea of what it might be like to search and navigate within linked data.

DBpedia
http://dbpedia.org

Freebase
http://www.freebase.com

Notes

1. "Library Linked Data Incubator Group Wiki," main page, W3C website, last modified February 22, 2012, www.w3.org/2005/Incubator/lld/wiki/Main_Page.
2. Tim Berners-Lee, "Linked Data," W3C website, July 27, 2006, last modified June 18, 2009, www.w3.org/DesignIssues/LinkedData.html.
3. Karen Coyle and Thomas Baker, "Guidelines for Dublin Core Application Profiles," W3C website, May 18, 2009, http://dublincore.org/documents/profile-guidelines.
4. Richard Cyganiak and Anja Jentzsch, "The Linking Open Data Cloud Diagram," last modified September 19, 2011, http://richard.cyganiak.de/2007/10/lod.
5. Anja Jentzsch,"DBpedia - Querying Wikipedia like a Database," Anja Jentzsch homepage on Freie Universität Berlin website, last modified February 10, 2012, www.wiwiss.fu-berlin.de/en/institute/pwo/bizer/team/JentzschAnja.html.