Making a Library Catalogue Part of the Semantic Web

Martin Malmsten
National Library of Sweden,
LIBRIS department, Sweden
martin.malmsten@kb.se

Abstract

Library catalogues contain an enormous amount of structured, high-quality data, however, this data is generally not made available to semantic web applications. In this paper we describe the tools and techniques used to make the Swedish Union Catalogue (LIBRIS) part of the Semantic Web and Linked Data. The focus is on links to and between resources and the mechanisms used to make data available, rather than perfect description of the individual resources. We also present a method of creating links between records of the same work.

Keywords: rdf; library catalogue; semantic web; linked data; persistent identifiers; frbr; sparql

1. Introduction

Even though bibliographic exchange has been a reality for decades, exchange of authority information and links between records are still not widely implemented. The standard way of making bibliographic data available is still through search-retrieve protocols such as SRU/W[32] or Z39.50[33]. Though this makes single bibliographic records retrievable, it does not provide a way to directly address them and reveals little or nothing about links between records. In contrast the Semantic Web (Berners-Lee et al., 2001) is by definition built upon linking of information. The promise of the Semantic Web and Linked Data (Berners-Lee 2006) is that it could make data connected, simply by making it available. This, it seems, could be the perfect way for libraries to expose all of their data.

A goal when creating the new version of the LIBRIS web interface[34] was to make the information presented to a normal user transparently available to machines/web robots as well. It was also obvious that information not intrinsic to the record itself, such as user annotations and connections to other records could be made available this way.

Also, thirty years of continually changing cataloguing rules and practices have left some data in an inconsistent state. Our hope is that the result of the work described will help us work with data in a new and better way.

2. Technical Overview

The Swedish Union Catalogue comprises about 175 libraries using a single Integrated Library System (ILS) for cataloguing. MARC21 is used for bibliographic, holdings and authority records. It contains about six million bibliographic records. A number of components were developed to make the ILS "talk RDF".

We created an RDF server wrapper to make the ILS accessible through HTTP and able to deliver RDF describing bibliographic and authority resources upon request, as well as RDF describing the links between them. Persistent URIs were created by using each record's unique number, these URIs can be dereferenced and will deliver the RDF when queried properly through HTTP content negotiation.

[32] Search/Retrieval via URL - http://www.loc.gov/standards/sru/

[33] Z39.50 - http://www.loc.gov/z3950/agency/

[34] LIBRIS - http://libris.kb.se/

https://doi.org/10.23106/dcmi.952109265

This data could then be loaded into a triple store to enable searching using SPARQL (Prud'hommeux and Seaborn, 2008).

3. Implementation

In this section we will outline the individual components of the implementation. A schematic is provided in FIG. 1.

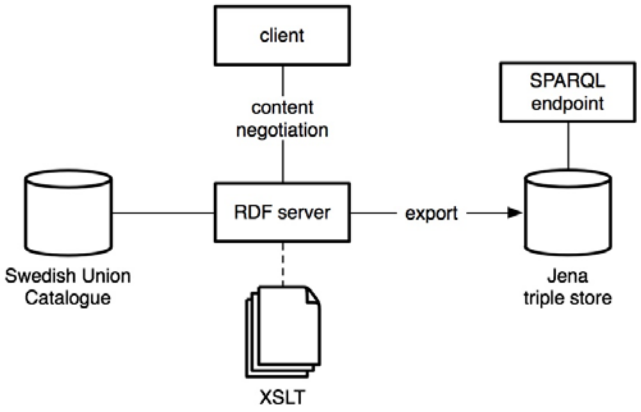

FIG. 1. Implementation schematic

3.1. RDF Server Wrapper

The first step was to create a wrapper around the ILS that could deliver the records in RDF rather than the binary format normally used for bibliographic records (ISO2709). The wrapper talks to the ILS using SQL and delivers records given its unique number. It then converts the ISO2709 record into an XML representation of MARC21. In the final step a transformation is applied to the XML using XSLT (Clark 1999).

Since each output format is implemented in a single XSLT-file, adding a new format or making changes to an existing one is trivial.

3.2. Linked Data and Access

Links and access are crucial underpinnings of both the semantic and "normal" web. For a resource to be linkable it needs a URI, for it to be accessible, that URI should be a HTTP one. Following the four rules of Linked Data (Berners-Lee 2006), a persistent, dereferenceable URI is created for each record. For bibliographic records: http://libris.kb.se/resource/bib/<number>, and for authority records: http://libris.kb.se/resource/auth/<number>.

Using HTTP content negotiation, the correct format can be delivered depending on the clients capabilities. This method uses the HTTP Accept header to tell the server what media types the client can handle and prefers. For example, the accept header text/html tells the server to deliver an HTML page suitable for a human user. An accept header containing, for example, text/rdf+n3 or application/rdf+xml tells the server that the client is able to handle RDF. The server can either deliver the data in RDF directly or send an HTTP 302 or 303 response indicating that the information can be found at a different URL (Sauermann, Cyganiak, 2008). See Appendix A for an example of content negotiation.

3.3. SPARQL Endpoint

We were interested in using SPARQL as a tool to both query and analyze data. Some queries that can be hard, or impossible, to formulate using SQL or a full text search language are easily formed using SPARQL. For example, the following query: "show me all subjects of records that belongs to the same work as the record with identifier XYZ". A query like this can be very useful

https://doi.org/10.23106/dcmi.952109265

for someone wanting to "auto complete" missing subject entries on records belonging to the same work. We used the Jena Semantic Web Framework[35] to create a triple store to hold the data. This gave us, with a minimum of work, the possibility to query data using SPARQL. A SPARQL endpoint conforming to the SPARQL Protocol for RDF (Clark, Feigenbaum, Torres, 2008) was implemented to allow queries over HTTP.

4. Types of Resources Described

There are a number of types of resources that needs to be described or made available to reflect the current state of a library catalogue, e.g books, authors, subjects (for controlled vocabularies and thesauri), organizations, links between them, etc. To make the library catalogue available to systems outside the library community, the resources should be described using common vocabularies. We used Dublin Core for bibliographic data, FOAF[36] for persons and organizations, and SKOS[37] for controlled vocabularies. These are all widely used and understood standards. An example graph is displayed in FIG. 2. See Appendix A for example records.

It is important to point out that it is possible to deliver multiple formats in parallel, so catering to the world outside the library community does not exclude systems aware of library standards. As described in 3.1 RDF Server Wrapper adding support for Bibliontology, MODS, MarcOnt or any other standard is easy, it is, however, not the subject of this paper.

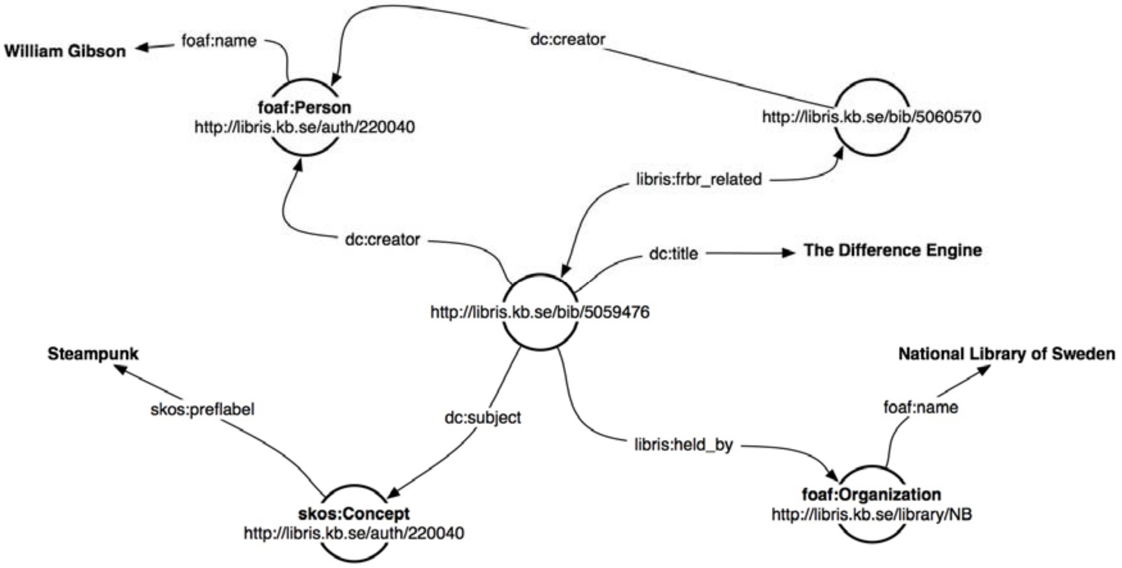

FIG. 2. Partial graph for the book "The Difference Engine"

5. FRBR

The Functional Requirements for Bibliographic Records (IFLA Section on Cataloguing, 1998) has been around for a decade, much has been written about it, though actual implementations are few. One hurdle to overcome is the shifting quality of the records due to continually changing practices. However, the idea of grouping or linking records being part of the same work is an appealing and technically viable one.

Every record in the LIBRIS database gets assigned one or more FRBR-keys, these keys are the normalized concatenations of an author and the original title. The process is repeated for each author and title. For example, the book "The Difference Engine" by William Gibson and Bruce

[35] Jena Semantic Web Framework - http://jena.sourceforge.net/
[36] Friend of a Friend - http://www.foaf-project.org/
[37] Simple Knowledge Organization System - http://www.w3.org/2004/02/skos/

https://doi.org/10.23106/dcmi.952109265

Sterling has two keys: "GIBSON WILLIAM 1948 THE DIFFERENCE ENGINE", and "STERLING BRUCE 1954 THE DIFFERENCE ENGINE". Links are then created between records with the same key.

This is similar to the approach of Styles et al. (2008) where the MD5 checksum of the name of the author and the title of the work are used as an identifier.

However, an important distinction compared to Styles et al. is that these keys are transient; they are never used as identifiers, only to create the links between records of the same work. This way, when an author dies, changes his/her name, etc. the links remain the same even though the keys change. There is therefore no need to keep track of changes since no identifier has been published. Another advantage is that works with more than one author is handled automatically, as well as records containing more than one work.

The LIBRIS database also contains actual work records in the form of name+title authority records. These are linked to their respective bibliographic records. The sheer amount of bibliographic records prohibits manual creation of these for the whole database, nevertheless these links are included in the RDF.

6. Links to External Resources

Linking to external resources gives the client a way of finding more information about a given resource. As a proof-of-concept the LIBRIS database contains a handful of links from authority records to DBpedia and Wikipedia. See Appendix A for an example.

We have also experimented with user annotation using the annotea ontology. Since the URIs used to identify records/resources are available outside the ILS, attaching data, such as user reviews, to them is easy and non-intrusive.

7. Conclusion

Although there are a number of ontologies available to describe bibliographic data, the data contained in library systems are not generally available. The access mechanisms described in Linked Data need to be implemented for libraries to truly be "part of the semantic web".

SPARQL shows real promise when it comes to mining the bibliographic data for information due to it's linked nature.

Planned next steps include using SPARQL for automatic creation of work records, implementing a richer description of bibliographic and authority records and loading more external data into the triple store. We are closely following the work of the DCMI/RDA Task Group[38].

We are currently exploring the possibility of making parts of this work available as Open Source. More information will be available at http://libris.kb.se/semweb.

[38] http://dublincore.org/dcmirdataskgroup/

https://doi.org/10.23106/dcmi.952109265

References

Berners-Lee, Tim, James Hendler, and Ora Lassila. (2001). The Semantic Web. *Scientific American, 284,* 34-43.

Berners-Lee, Tim. (2006). *Linked data*. Retrieved April 12, 2008, from http://www.w3.org/DesignIssues/LinkedData.html.

Prud'hommeaux, Eric, and Andy Seaborn. (2008). *SPARQL Query Language for RDF*. Retrieved April 12, 2008 from http://www.w3.org/TR/rdf-sparql-query/.

Clark, James. (1999). *XSL Transformations (XSLT)*. Retrieved April 13, 2008 from http://www.w3.org/TR/xslt.

IFLA Study Group on the Functional Requirements for Bibliographic Records. (1998). *Functional Requirements for Bibliographic Records*. Retrieved April 13, 2008 from http://www.ifla.org/VII/s13/frbr/frbr.pdf.

Styles, Rob, Danny Ayers, and Nadeem Shabir. (2008). Semantic MARC, MARC21 and the Semantic Web. *WWW 2008 17th International World Wide Web Conference*.

Sauermann, Leo, and Richard Cyganiak. (2008). *Cool URIs for the Semantic Web*. Retrieved April 13, 2008 from http://www.w3.org/TR/cooluris/.

https://doi.org/10.23106/dcmi.952109265

Appendix A. - Examples of HTTP Requests and Responses

The following are HTTP traces of requests for bibliographic and authority records.

1. Bibliographic record - request, redirect and response

```
GET /resource/bib/5059476
Host: libris.kb.se
Accept: text/rdf+n3
----------------------------------------
HTTP/1.1 303 See Other
Location: http://libris.kb.se/data/bib/5059476
----------------------------------------
GET /data/bib/5059476
Host: libris.kb.se
Accept: text/rdf+n3
----------------------------------------
HTTP/1.1 200 OK
Content-Type: text/rdf+n3

@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix libris: <http://libris.kb.se/experimental/> .
@prefix annotea: <http://www.w3.org/2000/10/annotation-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# RDF in Turtle/N3 created for the bibliographic record 5059476
<http://libris.kb.se/resource/bib/5059476>
       foaf:page <http://libris.kb.se/bib/5059476>;
       rdfs:isDefinedBy <http://libris.kb.se/data/bib/5059476>;

       # short bibliographic description
       dc:title          "The difference engine";
       dc:creator  "Gibson, William, 1948-";
       dc:creator  "Sterling, Bruce, 1954-";
       dc:subject  "Steampunk";
       dc:identifier <URN:ISBN:0-575-04762-3>;
       ...

       # links to authors with authority records
       dc:creator  <http://libris.kb.se/resource/auth/220040>;
       dc:creator  <http://libris.kb.se/resource/auth/307779>;

       # links to subjects with authority records
       dc:subject  <http://libris.kb.se/resource/auth/308073>;
       dc:subject  <http://libris.kb.se/resource/auth/308074>;

       # links to other editions of the same work
       libris:frbr_related <http://libris.kb.se/resource/bib/5060570>;

       # user annotations
       annotea:hasAnnotation <http://libris.kb.se/resource/annotation/123>;

       # book is held by the following libraries
       libris:held_by    <http://libris.kb.se/resource/library/Sk>;
       libris:held_by    <http://libris.kb.se/resource/library/Vvt> .
```

2. Authority Record for Author William Gibson - Request and Response

```
GET /data/auth/220040
Host: libris.kb.se
Accept: text/rdf+n3
----------------------------------------
HTTP/1.1 200 OK
Content-Type: text/rdf+n3

@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dbpedia: <http://dbpedia.org/property/> .
```

https://doi.org/10.23106/dcmi.952109265

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# RDF in Turtle/N3 created for the authority record 220040
<http://libris.kb.se/resource/auth/220040>
       rdfs:isDefinedBy <http://libris.kb.se/data/auth/220040>;

 # type of authority record
 rdf:type<http://xmlns.com/foaf/0.1/Person> ;

 # description
 foaf:name "Gibson, William, 1948-" ;
 foaf:name "William Gibson" ;
 ...

 # links to external resources
 owl:sameAs  <http://dbpedia.org/data/William_Gibson> ;
 rdfs:seeAlso <http://en.wikipedia.org/wiki/William_Gibson> .

# links to books by this author
<http://libris.kb.se/resource/bib/2716178>                            dc:creator
<http://libris.kb.se/resource/auth/220040> .
<http://libris.kb.se/resource/bib/2793076>                            dc:creator
<http://libris.kb.se/resource/auth/220040> .
<http://libris.kb.se/resource/bib/4465470>                            dc:creator
<http://libris.kb.se/resource/auth/220040> .
<http://libris.kb.se/resource/bib/4574314>                            dc:creator
<http://libris.kb.se/resource/auth/220040> .
...
```

3. Authority Record for the Subject Steampunk - Request and Response

```
GET /data/auth/308074
Host: libris.kb.se
Accept: text/rdf+n3
----------------------------------------
HTTP/1.1 200 OK
Content-Type: text/rdf+n3

@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .

# RDF in Turtle/N3 created for the authority record 308074
<http://libris.kb.se/resource/auth/308074>
       rdfs:isDefinedBy <http://libris.kb.se/data/auth/308074>;

       # type of authority record
       rdf:type         skos:Concept ;

       # description
       skos:prefLabel  "Steampunk" ;
       skos:related    "Science fiction" ;
       skos:related    "Cyberpunk" ;

       # links to other subjects
       skos:related    <http://libris.kb.se/resource/auth/243892> ;
       skos:related    <http://libris.kb.se/resource/auth/142481> ;

       # links to external resources
       owl:sameAs <http://dbpedia.org/page/Steampunk> .

# links to books with this subject
<http://libris.kb.se/resource/bib/5059476>                            dc:subject
<http://libris.kb.se/resource/auth/308074> .
<http://libris.kb.se/resource/bib/5060570>                            dc:subject
<http://libris.kb.se/resource/auth/308074> .
```

https://doi.org/10.23106/dcmi.952109265