Seeding Structured Data by Default via Open Source Library Systems

Dan Scott

Laurentian University,
Sudbury, Ontario, Canada
dscott@laurentian.ca
https://coffeecode.net

Abstract. Most libraries use the machine-readable cataloguing (MARC) format to encode and exchange metadata about the items they make available to their patrons. Traditional library systems have not published this data on the Semantic Web. However, some agile open source library systems have begun closing this gap by publishing structured data that uses the schema.org vocabulary to describe the bibliographic data, make offers for items available for loan, and link the items to their owning libraries. This article distills the lessons learned from implementing structured data in Evergreen, Koha, and VuFind; highlights emerging design patterns for publishing structured data in other library systems; and traces the influence these implementation experiences have had on the evolution of the schema.org vocabulary. Finally, we discuss the impact that "the power of the default" publishing of structured data could have on discoverability of library offerings on the Semantic Web.

Keywords: #eswc2014Scott, Libraries, Structured data, MARC formats, schema.org, Open source

1 Introduction

A pragmatic incentive for publishing structured data on the Semantic Web is the promise that elevating web pages beyond mere bags of words will enable search engines to provide better responses to queries through strategies such as disambiguating terms. Search engines have assumed the most visible role of the intelligent agents described in Berners-Lee's seminal vision of the Semantic Web [1]. The goals of many search engine users parallel the information seeking goals Nardi originally classified for users of libraries, such as monitoring, planned, or exploratory searches [2]. As non-commercial repositories of resources that can satisfy these classes of queries, libraries have continually designed and evolved organizational systems for indexing and efficiently locating their resources (*traditional library systems*, as used in almost every public, academic, and special library in many parts of the world). This paper explores the results of enabling open source instances of these systems to participate in the Semantic Web by adopting structured data conventions and the generalist schema.org vocabulary,

as encouraged by major search engines, in what is commonly thought to be a highly specialized domain.

To facilitate remote queries to traditional library systems, libraries were early adopters of technology such as direct dial-up connections, TELNET, and web catalogues [3]. Since the 1960s, most traditional library systems have used the Machine Readable Cataloguing (MARC) record format to describe and exchange metadata about their resources. Traditional library systems were therefore well-positioned to be early participants in the Semantic Web.

However, cataloguing practices as encoded by MARC records have focused on strings, not things; and the strings themselves often described more than a single property. For example, until 2013, the MARC 21 format used by North American libraries combined ISBNs and a description of the physical format for the book holding that ISBN in a single descriptive field without standardized delimiters [4]. In the field of linked data, most traditional library systems offer rudimentary support for creating and maintaining links between entities such as author names by relying on *authority records* that are maintained within the same system, but those inward-facing links have not been exposed as structured data on the web.

2 Libraries at an Impasse

Where libraries have made efforts to expose the raw metadata of their traditional library systems in a more machine-readable way, adoption has been uneven and these approaches are not well-known outside of library or bibliographic contexts. For example, the COinS microformat [5] encodes NISO Z39.88 metadata in an OpenURL ContextObject that can be used to cite and locate a copy of the work. The unAPI microformat [6] enables client applications to retrieve raw metadata records in different formats such as MARCXML, MODS, and RIS. However, while these methods of surfacing machine-readable metadata are consumed by client applications such as Zotero and Mendeley, they solve specific bibliographic problems rather than broader Semantic Web problems.

By late 2011 it was clear that most traditional library systems were being left out of the emerging Semantic Web. Google continued to push for the adoption of structured data [7], and then joined Yahoo! and Bing in unveiling the schema.org vocabulary [8] with promises that web pages using schema.org could receive special treatment from search engines when it came to display ("rich snippets") and relevance. Summers succintly summarized the problem faced and caused by libraries, stating "the use of HTML5 Microdata and schema.org by Google, Bing and Yahoo, and the use of RDFa by Facebook are [...] good reminders that the library software development community is best served by paying attention to mainstream solutions, as they become available, even if they eclipse homegrown stopgap solutions" [9].

While thought leaders such as the Swedish Union Catalogue (LIBRIS) [10], the Deutsche National Bibliothek[11], the Bibliothèque nationale de France [12], and OCLC have all implemented linked data patterns, those initiatives occurred

in highly centralized organizations and their results are not easily replicated by libraries with less concentrated development resources. Ronallo's analysis of the August 2012 Common Crawl corpus found that American academic library sites had failed to respond to Summers' challenge, as those sites contained very few schema.org instances: a mere 8,351 instances of Article, 1,275 instances of CollectionPage, and 298 instances of ScholarlyArticle represented the most common academic types [13].

Unfortunately, those smaller libraries that would be willing to contribute their data to the Semantic Web [1] generally lack the resources necessary to customize their existing systems, pressure or incentive vendors to enhance their software, or invest in difficult transitions to new systems. Budget challenges in particular force administrators to focus on more mundane efforts such as collection development and hinder efforts that are not perceived as offering immediate results for their users.

3 Open Source Library Systems

In a risk-averse, static domain, open source library systems offer hope for regular libraries. Many proprietary systems allow client libraries to suggest and vote on the prioritization of development efforts, but those results are not binding, and the pool of available development resource is limited to a single vendor. In comparison, the communal development effort for open source software means that a given enhancement needs to only be developed once, then shared with all other users of the same system; and "[d]istribution of source code can lead to efficiency gains by making it possible for the modifications to be done by those actors who have the best information about their value [and] are best equipped to carry them out" [14].

Accordingly, we hypothesized the simplest and most effective solution to increasing the amount of structured data published by libraries was to enhance open source library systems so that they would publish structured data by default. Just as "getting the default "right" could have a tremendous impact on the distribution of retirement savings available to individuals" [15], we felt it was important that library systems should start publishing structured data as soon as they were installed or upgraded. If libraries had to opt in to publishing their data through a configuration setting, or had to make minimal customizations to the web layer of their systems, then a significant portion of libraries would *not* choose to opt in; even if the option was found, the decision to change the default would be complex for those who are not experts with the Semantic web. Per Madrian, "the default will assume an asymmetric position in the decision-making process relative to other outcomes, and consequently, will be more likely to be picked as the chosen alternative" [15].

When working towards implementing linked data principles [16] in open source library systems, we have the advantage that all of these systems are native to the web and did not evolve from pre-web networks. For example, while many proprietary systems still use session parameters as part of their URL scheme and

thus break basic functionality like bookmarking or sharing URLs, open source library systems such as Koha, Evergreen, and VuFind all use persistent URIs to offer access to individual records. This satisfies the linked data requirements to use URIs to name things (at least at the level of individual record) and to use dereferenceable HTTP URIs.

Our efforts to enable library systems to publish structured data by default focused on two mature open source library systems (Koha and Evergreen), and one mature open source discovery system: VuFind.

3.1 Evergreen

In 2004, the Georgia Public Library Service (GPLS) decided to fund the development of the Evergreen library system because "[t]he limit reached [by the existing software] was a hard one, and there was no solution using that software. Meanwhile, more libraries wanted to join PINES [the consortial resource sharing library system]" [17]. In 2006, the first version of Evergreen was released under the GPL version 2 licence and GPLS PINES launched with 252 libraries running under a single Evergreen instance. Evergreen is now used by at least 1,388 libraries worldwide [18]. We included Evergreen due to its broad reach and its familiarity to the author of this paper, who has been an Evergreen developer since 2007.

3.2 Koha

Koha was developed in 1999 to replace a proprietary system that suffered from severe Y2K-compliance problems and for which the company no longer existed. Development of the initial version of Koha was funded by a single library, the Horowhenua Library Trust (HLT), who opted to release the software under an open source license as a "gift given freely" (the meaning of the Maori word "Koha") [19]. From that single library in 2000, Koha is now being used by at least 2,500 libraries worldwide [20]. As the most widely adopted open source library system with the most mature development team and process, we felt that including Koha in our implementation efforts would have a significant impact.

3.3 VuFind

Rather than replacing its proprietary library system entirely, Villanova University opted to instead build a new discovery layer that could blend the results of both the library system catalogue and other sources such as article databases and an institutional repository of theses and dissertations. The resulting software, VuFind, began development in 2007, reached a 1.0 release in 2010 [21], and has continued to iterate with a small but robust development team. There are 135 self-reported installations [22], including York University, who listed seven key criteria in their decision to implement VuFind [23]. York's criteria did not include Semantic Web considerations, but we included VuFind in our implementation efforts because their small but stable community is growing, their

development team was amenable to the addition of structured data, and we looked forward to the challenge of working with data from disparate systems.

4 Mapping Library System Records to the schema.org Vocabulary

For the purposes of this paper, publishing library system records on the web using schema.org structured data required three steps:

1. Determining the schema.org type of the bibliographic record
2. Mapping the record elements to the type's properties
3. Linking physical or electronic resources to the described object

4.1 Determining the schema.org Type of the Bibliographic Record

Koha and Evergreen expose raw MARC records to their display templates, so to properly determine their schema.org types we need to analyze both the MARC leaders and the fixed fields to discern the type of the described bibliographic data. To complicate matters, Koha can support both the MARC 21 and UNIMARC formats, each of which features their own rules for encoding bibliographic information. For example, determining that a given MARC 21 record describes a motion picture requires us to check the 6th character of the record leader, then check the 33rd character of the 008 field. As Evergreen supports only MARC 21, we narrowed the scope of our efforts by focusing only on mapping MARC 21 records, and due to the complexity of MARC 21 format, only map `schema:Book`, `schema:Map`, and `schema:MusicAlbum`, with a fallback to `schema:CreativeWork`, for this initial effort.

In contrast, VuFind supports the creation of a single index sourced from heterogeneous sets of records including, but not limited to, MARC 21 records, by normalizing the source records to a common, simplified schema. While the simplified schema inhibits us in some cases from publishing structured data properties as granular as when we have access to the raw MARC 21 records, its strictures liberate us from having to craft intricate mappings of the raw data. Therefore, in addition to the mappings available to us in Evergreen and Koha, in VuFind we were easily able to also map records to `schema:Movie` and `schema:Photograph`.

The following table lists the schema.org types that we mapped, using the sixth character of the MARC 21 leader as a guide:

Table 1: MARC 21 leader[06] values to schema.org types

Schema.org type	**MARC 21 leader[06] value**
Book	a
Map	e
MusicAlbum	j
CreativeWork	All other LDR values

Articles (which would map to `schema:Article`) fall under the "Language material" designation used by books, and individual music tracks (which would map to `schema:MusicRecording`), fall under the "Musical sound recordings" designation used by albums. However, neither articles nor individual tracks are typically described in these library systems and were excluded from this research.

4.2 Mapping the Record Elements to the Type's Properties

Once we have mapped the record to a schema.org type, we can map the record elements to the properties for the type. As the base types all inherit from `schema:CreativeWork`, common properties such as `schema:author`, `schema:contributor`, `schema:name`, `schema:datePublished`, and `schema:publisher` can be mapped once and reused for all types. Following this approach, special handling is required only for the extended properties for types such as `schema:MusicAlbum` which, rather than `schema:author` or `schema:contributor`, prefers a `schema:byArtist` property with a range of `schema:MusicGroup`.

The following table describes the mappings to schema.org properties from combinations of MARC 21 fields and their subfields. Unless otherwise indicated, the values of all subfields for a given occurrence of a field were concatenated to provide the value for a single occurrence of a schema.org property. "CreativeWork" implies all schema.org children, such as Book, Map, and MusicAlbum. *Note*: `schema:birthDate` and `schema:deathDate` are derived from the same subfield using the supplied regular expression.

Table 2: MARC 21 field/subfield values to schema.org properties

Schema.org property	**MARC 21 field/subfield**
CreativeWork/name	`245/All subfields except w, 0, 4, 5, 6, 8, 9`
Book/isbn	`022/a`
CreativeWork/publisher/Organization/location	`(260/a or 264[indicator 2="1"])/a`
CreativeWork/publisher/Organization/name	`(260/b or 264[indicator 2="1"])/b`
CreativeWork/datePublished	`(260/c or 264[indicator 2="1"])/c`
CreativeWork/keywords	`(600, 610, 611, 630, 650, 651, 655, 659, 690, 692, 693, 698, 699)/a-z`

MusicRecording/byArtist/MusicGroup/name	`110/a-z`
CreativeWork/author/Person/name	`100/a-z`
CreativeWork/author/Organization/name	`(110, 111)/a-z`
CreativeWork/contributor/Person/name	`700/a-z`
CreativeWork/contributor/Organization/name	`(710, 711)/a-z`
CreativeWork/author/Person/birthDate	`100/d '^\s*(\d{4}).*$'`
CreativeWork/author/Person/deathDate	`100/d '^\s*.{4}-(\d{4}).*$'`

4.3 Linking Physical or Electronic Resources to the Described Object

Libraries make specific resources available for use, so simply describing the general resource is not sufficient. Semantic Web agents need to be able to determine which library holds the resource, where the resource is located within the library, and whether it is available. We discuss this in detail as one of the emerging design patterns in the following section.

5 Emerging Design Patterns

Following a tactic of first marking up the text as it already exists on the web page, our initial implementation efforts simply published the personal and corporate names, as given in the source data, as literal values for the `schema:contributor` property. While for schema.org it "is not a requirement [to satisfy the expected range of a given property]–it's fine to include just regular text or a URL" [24], as libraries we strive to publish high quality structured data. Several notable design patterns emerged through our efforts to enable library systems to publish rich schema.org structured data.

Providing Better Granularity for Personal and Corporate Names: To distinguish common personal names from one another, MARC 21 records may include the birth date and death date (if applicable) in a single undifferentiated subfield. Regular expressions enable us to disambiguate that data into separate `schema:birthDate` and `schema:deathDate` properties, thus enriching the structured data that we publish beyond what the source record explicitly encoded. We were also able to differentiate between corporate authors, individual authors, and contributors to works, as well as provide special handling for music groups, rather than indiscriminately adding `schema:author` properties to works.

Linking Resources to the Described Object: One of the core functions of library systems is to serve as a catalogue that enables users to locate items and determine the current status of those items; it effectively serves as a highly localized search engine. A previous iteration of schema.org structured data in Evergreen resulted [25] in only 3,275 `schema:Book` instances being reported by Google's Webmaster Tools out of what should have been hundreds of thousands,

and had no discernible impact on searches in Google or Bing in informal testing of the catalogue. Based on these results, we hypothesized that search engines want to connect searchers directly to the items that they are seeking; therefore, when we publish structured data, we now use the `schema:Offer` type to expose copies of resources. As the defined range of the `schema:itemOffered` property is `schema:Product`, we use multiple types in the RDFa `@typeof` property to express both `schema:Product` and the appropriate `schema:CreativeWork` (or child type such as `schema:Book`). This enables us to use properties from both types to describe both the generic object and the offer-specific attributes, as follows:

Table 3: Mapping available resources to the described object

Schema.org type or property	**Library entity**	**Notes**
CreativeWork/offers/Offer	Holding, item, or copy	Repeated once per holding
Offer/businessFunction	Borrowing terms (for example, reserved for in-library use)	Available for loan = http://purl.org/goodrelations/v1#LeaseOut
Offer/itemOffered	Bibliographic record	
Offer/sku	Call number or shelf mark	As a literal "stock keeping unit" number, call number shares the properties of enabling the location of a group of copies of items using a single number.
Offer/seller/Library/name	Library name	Koha and VuFind use the literal value of the library name, while a working branch in Evergreen offers a full `schema:Library` object (see below).
Offer/serialNumber	Barcode	Satisfies the need for a unique identifier for an individual copy.
Offer/gtin13	ISBN	
Offer/availableAtOrFrom	Shelving location	Currently mapped to the literal value of the name of the shelving location (for example, "Stacks"), but finer granularity could be achieved through the use of `schema:containedIn`.
Offer/description	Public copy notes	

We mapped the availability of resources from common library terminology to the `schema:ItemAvailability` enumeration as follows:

Table 4: Mapping `schema:ItemAvailability` to library resource availability

schema.org type	Type of availability
`schema:InStock`	Available on shelf or awaiting reshelving
`schema:OutOfStock`	Checked out or waiting to be picked up for a hold
`schema:PreOrder`	On order, in process, or in transit to another library
`schema:InStoreOnly`	Reserved for on-site usage

Linking Resources to the Offering Library: The "seller" property of `schema:Offer` has a formally defined RDF range of `schema:Organization` or `schema:Person`; however, in keeping with the pragmatic nature of schema.org, our initial implementations simply supplied a non-semantic literal–the name of the library–or linked to an external web page that, as it is out of our control, at this time most likely does not include any structured data.

A prototype implementation in Evergreen [26] generates one web page per library containing structured data based on the `schema:Library` type. As an RDF subclass of `schema:Organization`, `schema:Library` satisfies the range constraints of `schema:seller`, and it offers an expressive set of properties to describe the organization such as contact information and hours of operation. Most library systems must maintain a current set of library operating hours to avoid accruing fines during closed times, and manage contact information such as email addresses, phone numbers, and mailing addresses to facilitate communication with users. Our enhancement generates data-rich `schema:Library` web pages that not only support Semantic Web needs, but also offer value to users and libraries by surfacing some of the most important library information directly from the relational database underpinning Evergreen.

6 Extending the schema.org Vocabulary

The author of this paper has had the pleasure of working closely with the schema.org community directly via the W3 Web Schemas group [27] and indirectly through the W3 Schema.org Bibliographic Extensions Community Group (*SchemaBibEx*) [28]. These collaborative efforts have led to several enhancements of the schema.org vocabulary.

6.1 Decommercializing the `schema:Offer` and `schema:Product` Types

When schema.org incorporated the GoodRelations vocabulary for the `schema:Product` and `schema:Offer` types, it simplified the core type and property descriptions by deemphasizing the generic agent-promise-object model underpinning the GoodRelations ontology [29] and focused instead on commercial transactions for the primary use case in schema.org. However, the `schema:Offer` definition of "An offer to sell an itemfor example, an offer to sell a product, the DVD of a movie, or tickets to an event" [30] failed to encompass many of the services offered by

libraries or other non-commercial entities. When we proposed the holdings-as-Offer pattern to SchemaBibEx, several participants raised objections due to the commercial nature of the existing Offer documentation that was thought to be unsuitable for a library context. Accordingly, we proposed changes to the definitions of three types, three properties, and 11 enumerated values such that they would also accommodate non-commercial transactions and services. The proposal was accepted with minor improvements and is scheduled to be incorporated into the next revision of both the schema.org vocabulary and the GoodRelations vocabulary [31, 32].

6.2 Establishing Clear Usage Patterns for Multiple Types

schema.org users have repeatedly expressed confusion about the appropriate usage of multiple types in microdata and RDFa [34–36]. The emerging consensus that multiple schema.org types can be expressed in a single microdata `@itemprop` or RDFa `@property` attribute, while additional types from outside the schema.org vocabulary should be expressed via a separate `schema:additionalType` property (for microdata) or in the same `@property` attribute (for RDFa), reflects the usage pattern we adopted for linking library resources to their described objects. Our work has served as a practical example in answers to these questions.

6.3 Extending the Vocabulary to Encompass Magazines, Journals, and Other Periodicals

Although the proposal has not yet received final approval, one of the recent SchemaBibEx efforts has been to define a set of schema.org types and properties that would enable libraries and publishers to describe periodicals at the title and issue level. Given the existence of the `schema:Article` and `schema:ScholarlyArticle` types and the `schema:citation` property, there is a strong need to be able to express the publication and issue in which an article has been collected. One could use a separate vocabulary such as the Bibliographic Ontology [33], but that runs counter to the schema.org goal of providing "a single place to go to learn about markup, instead of having to graft together a schema from different sources, each with its own rules, conventions and learning curves" [38]. Therefore, while informed by the existing Bibliographic Ontology work in this area, the current proposal [37] hews closer to the existing `schema:Series` / `schema:Season` / `schema:Episode` pattern by promoting volume (a collection of issues, typically by year) into a first-class type.

7 Discussion

7.1 Assessing schema.org for Traditional Library Systems

Although previous Semantic Web efforts for library systems adopted a mix of multiple specialized vocabularies such as FOAF, SKOS, and Dublin Core [10–12], we chose to assess the ability of the generalist schema.org vocabulary to

satisfy the needs of the library domain. In theory, using a single vocabulary to express structured data should simplify the publishers' mapping effort and ease the consumers' ability to consume the data. Given that the major search engines have endorsed schema.org, use of that vocabulary is expected to increase the visibility of library resources in search engines. By limiting ourselves to the schema.org vocabulary, we are well-positioned to test these perceived advantages as libraries using Evergreen, Koha, and VuFind upgrade to the schema.org-enhanced versions of the software.

While mapping human-visible elements of web catalogues to schema.org structured data, we observed that traditional library systems *could* usefully deploy schema.org and achieve an acceptable level of metadata granularity through the previously described mapping design patterns. We also identified several cases in which the schema.org enhancement process successfully addressed gaps identified by SchemaBibEx. Current proposals such as MiniSKOS [39] promise to address the longer tail of bibliographic description needs.

Although individual libraries can customize their own web catalogues on a site-by-site basis, developers of traditional library systems hold the potential to most efficiently bring libraries to the Semantic Web by enhancing their systems to publish structured data by default. The author's experience in successfully augmenting three separate library systems to publish schema.org structured data–including all code contribution, review, and integration processes–in approximately three months suggests that the implementation cost for developers of other systems should be relatively low, particularly given that the code from the work described by this paper is open for inspection and adoption.

7.2 Potential Impacts

If Evergreen, Koha, and VuFind are only the first of many library systems to publish schema.org structured data by default, we can speculate about some potential impacts a broader adoption of this approach may have in a library context:

Improved efficiency and accuracy of resource-sharing systems: To participate in the resource-sharing networks that support interlibrary loan services, libraries periodically deliver batch updates of their records and holdings or maintain a Z39.50 server to participate in a federated search system. Batch updates enable a central service to assemble a collection of all records held by the resource-sharing participants, but those records are outdated almost immediately as resources are added or removed on a daily basis at most libraries. Z39.50 is a complex library-specific search protocol that still suffers from the implementation inconsistencies cited by Lunau [40].

Given a set of participating libraries that publish structured data with agreed-upon schema.org mappings and sitemaps, however, a new centralized service could avoid manual batch update processes by instead periodically crawling all of the new and changed pages of its member libraries to maintain a centralized database. When a resource is requested, the availability could be checked

by requesting and parsing the resource page for `schema:Offer` entities with an agreeable `schema:itemAvailability` value.

Improved efficiency of–or disintermediated–OCLC: OCLC has emerged as one of the centralized entities responsible for mediating library-library interactions such as collaborative cataloguing efforts and resource sharing initiatives in North America, as well as supplying search engines such as Google with the data required to connect searchers to libraries [41]. Libraries currently make their resources known to OCLC (and thus to other libraries) through cumbersome "batch loads". Given schema.org structured data that follows the holdings-as-Offer pattern, OCLC could instead follow the approach established by search engines of using sitemaps to crawl library catalogues and update their indexes accordingly. With an even broader adoption of structured data by libraries, however, regular search engines could simply parse the available structured data from known libraries and return more relevant customized results based on signals such as the searcher's geographic location, known library preferences, and participation in social networks, effectively disintermediating OCLC from its current role as a metadata supplier to Google and other search engines.

7.3 Future Work

Future possibilities for work in this area include:

Improve the mapping from MARC 21 to schema.org, and create a mapping for UNIMARC: For MARC 21, mappings for the base types would benefit significantly from including the MARC 21 008 and 006 fixed fields in the analysis. The mapping of MARC 21 subfields to `schema:Person` and `schema:Organization` names should only include recognizable "name" values in the `schema:name` property, while other values can be directed to more appropriate properties or ignored.

Many Koha sites use the UNIMARC record format, which currently has no mapping for schema.org. While lossy conversions from UNIMARC to MARC 21 are available, a direct mapping from UNIMARC to schema.org may provide better structured data. Alternately, the use of a more semantic intermediary format such as the Metadata Object Description Schema (MODS) [42] may be a fruitful avenue of exploration for source formats including MARC 21 and UNIMARC. Shared documentation and implementations of these mappings would enable other library systems to benefit from a common analysis, assuming that they were available under an open source license, and would contribute to enhancing the contributions of libraries to the Semantic Web.

Broader implementation of the agent-promise-object model: The `schema:Offer` pattern for relating resources to the libraries that hold them using `schema:Library` to fulfill the agent-promise-object model has been prototyped in Evergreen [26]. To further the goal of structured data by default, we plan to extend this implementation to Koha and other library systems that track library locations, hours of operation, and contact information.

Link to external data: While we were able to publish structured data with persistent URIs, all links other than electronic resource URIs were siloed within

the library system. For MARC-based systems, the next step is to follow the existing conventions for linking bibliographic fields such as authors and subjects to authority records, and in turn link from the authority records to external records such as the Library of Congress Linked Data Service [43] or VIAF: The Virtual Authority File [44]. However, many MARC 21 fields–such as publication information recorded in MARC 260 or 264 fields–are not allowed to include linking subfields, and thus limits the basic MARC mapping approach to a best-effort string-matching approach.

Assess the growth of library-published structured data: The impact of these changes to three major open source library systems on the proliferation of structured data published by libraries needs to be assessed. Repeating Ronallo's Common Crawl analysis with a data set in one year's time should demonstrate the results (if any) of the "structured data by default" releases of Koha, Evergreen, and VuFind. Such a study should extend its scope beyond American academic libraries, and should annotate the results by which software published the structured data to provide insight into the impact of the subject of this paper.

Assess the impact of library-published structured data on users: We need to confirm the hypothesis that publishing structured data will have a tangible, positive impact for users of general search engines by running usability studies. Given Evergreen's ability to surface full Product-Offer-Library relationships, we expect that search engines should be able to tailor results to local users by directly including local library resources. To assess this hypothesis, a longitudinal usability study that compares user frustration and source of discovery (for example, library catalogue or general search engine) for a set number of known local resources at the start and end of the study may offer a fruitful approach.

8 Conclusion

In this paper we described the process and lessons learned from enabling the open source library systems Evergreen, Koha, and VuFind to publish schema.org structured data by default; highlighted several of the areas where this implementation experience affected the evolution and usage of the schema.org vocabulary; and discussed the potential impact "the power of the default" publishing of schema.org structured data can have on libraries. Our next steps are to refine and expand the implementations, to link out to external data, and to assess the impact of our efforts once libraries begin publishing structured data by default.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. Scientific American. 284(5), 28–37 (2001)
2. Nardi, B.A., O'day, V.: Intelligent agents: what we learned at the library. Libri. 46(2), 59–88 (1996)

3. Grothkopf, U.: The Internet for Librarians. Vistas in Astronomy. 39(2), 137–143 (1995)
4. 020 - International Standard Book Number (retrieved January 12, 2014), http://www.loc.gov/marc/bibliographic/bd020.html
5. OpenURL COinS: A Convention to Embed Bibliographic Metadata in HTML (retrieved January 12, 2014), http://ocoins.info/
6. UNAPI: An un-API for Webapps (retrieved January 12, 2014), http://unapi.info
7. Jordan, T.: Scalable Structured Markup. In: Google I/O 2011 (retrieved January 12, 2014). http://www.google.com/events/io/2011/sessions/scalable-structured-markup.html
8. Goel, K., Gupta, P.: Introducing schema.org: Search engines come together for a richer web (retrieved January 12, 2014), http://googlewebmastercentral.blogspot.ca/2011/06/introducing-schemaorg-search-engines.html
9. Summers, E.: GoodReads microdata (retrieved January 12, 2014), http://inkdroid.org/journal/2011/08/02/goodreads-microdata
10. Malmsten, M.: Making a Library Catalogue Part of the Semantic Web. In: Proceedings of the International Conference on Dublin Core and Metadata Applications, pp. 146–152. (2008)
11. Linked Data Service of the German National Library, http://www.dnb.de/EN/lds
12. Simon, A., Wenz, R., Michel, V., Di Mascio, A.: Publishing Bibliographic Records on the Web of Data: Opportunities for the BnF (French National Library). In: Cimiano, P., Corcho, O., Presutti, V., Hollink, L., Rudolph, S. (eds) ESWC 2013. LNCS, vol. 7882, pp. 563–577. Springer, Heidelberg (2013)
13. Ronallo, J.: Embedded Semantic Markup, schema.org, the Common Crawl, and Web Data Commons: What Big Web Data Means for Libraries and Archives. In: Digital Library Federation Forum 2013. http://jronallo.github.io/presentations/2013-dlf/presentation_with_notes/ (2013)
14. Schwarz, M., Takhteyev, Y.: Half a Century of Public Software Institutions: Open Source as a Solution to HoldUp Problem. Journal of Public Economic Theory. 12(4), 609–639 (2010)
15. Madrian, B.C., Shea, D.F.: The power of suggestion: Inertia in 401 (k) participation and savings behavior. The Quarterly Journal of Economics. 116(4), 1149–1187 (2001)
16. Bizer, C., Heath, T., Berners-Lee, T.: Linked data–the story so far. International Journal on Semantic Web and Information Systems. 5(3), 1–22 (2009)
17. Molyneux, R. Evergreen in Context. Bulletin of the American Society for Information Science and Technology. 35(2), 26–30 (2009)
18. lib-web-cats: A directory of libraries throughout the world (Evergreen) (retrieved January 12, 2014), http://www.librarytechnology.org/libraries.pl?ILS=Evergreen
19. Cormack, C., Poulain, P.: Kohacon 2009 Keynote. In: KohaCon 2009 (retrieved January 12, 2014). https://archive.org/details/Kohacon09Keynote (2009)
20. lib-web-cats: A directory of libraries throughout the world (Koha) (retrieved January 12, 2014), http://www.librarytechnology.org/libraries.pl?ILS=Koha
21. VuFind Change Log (retrieved January 12, 2014), https://vufind.org/wiki/changelog
22. VuFind Installations (retrieved January 12, 2014), https://vufind.org/wiki/installation_status
23. Denton, W., Coysh, S.J.: Usability testing of VuFind at an academic library. Library Hi Tech. 29(2), 301–319 (2011)

24. schema.org: Getting started with schema.org (retrieved January 12, 2014), http://schema.org/docs/gs.html#schemaorg_expected
25. Scott, D.: Microdata: making metadata matter. In: Evergreen International Conference (retrieved January 12, 2014). http://zone.biblio.laurentian.ca/dspace/handle/10219/1993 (2013)
26. Scott, D.: Add per-library TPAC pages with schema.org structured data support (retrieved January 13, 2014), https://bugs.launchpad.net/evergreen/+bug/1261939
27. W3C Web Schemas (retrieved January 12, 2014), http://www.w3.org/wiki/WebSchemas
28. W3C Schema Bib Extend Community Group (retrieved January 12, 2014), http://www.w3.org/community/schemabibex/
29. Hepp, M.: Goodrelations: An ontology for describing products and services offers on the web. In: Gangemi, A., Euzenat, J. (eds) EKAW 2008. LNCS, vol. 5268, pp. 329–346. Springer, Heidelberg (2008)
30. schema.org: Thing / Intangible / Offer (retrieved January 12, 2014), http://schema.org/Offer
31. Broaden Offer usage (retrieved January 12, 2014), http://www.w3.org/community/schemabibex/wiki/Broaden_Offer_usage
32. Scott, D.: Re: Support non-commercial usage of schema.org/Offer - RDF(S) patch (retrieved January 12, 2014), http://lists.w3.org/Archives/Public/public-vocabs/2013Dec/0042.html
33. D'Arcus, B., Giasson, F.: Bibliographic Ontology Specification (retrieved January 12, 2014), http://bibliontology.com
34. Bang, C.: CreativeWork can't be a Product? (retrieved January 12, 2014), http://lists.w3.org/Archives/Public/public-vocabs/2013Oct/0091.html
35. Scott, D.: Re: Proposal: Audiobook (retrieved January 12, 2014), http://lists.w3.org/Archives/Public/public-vocabs/2013Sep/0205.html
36. Deering, D.: Google+ Semantic Search Marketing community post (retrieved January 12, 2014), https://plus.google.com/107534960995428499496/posts/JeER7ugmRpf
37. Proposal for Periodicals, Articles and Multi-volume Works (retrieved January 12, 2014), http://www.w3.org/community/schemabibex/wiki/Article
38. schema.org: Frequently Asked Questions (retrieved January 12, 2014), http://schema.org/docs/faq.html#2
39. Brickley, D.: MiniSKOS proposal: add Topic, an equivalentClass to skos:Concept, and relate schema properties to it. (retrieved March 5, 2014), http://lists.w3.org/Archives/Public/public-vocabs/2013Nov/0121.html
40. Lunau, C.: The Virtual Canadian Union Catalogue Project (vCuc). Resource Sharing & Information Networks. 14(2), 21–35 (1999)
41. OCLC and Google to exchange data, link digitized books to WorldCat (retrieved January 12, 2014), http://worldcat.org/arcviewer/2/OCC/2010/05/07/H1273247173434/viewer/file327.htm
42. Library of Congress: Metadata Object Description Schema (MODS) (retrieved January 12, 2014), http://www.loc.gov/standards/mods/
43. Library of Congress: Linked Data Service (retrieved January 12, 2014), http://id.loc.gov
44. VIAF: The Virtual International Authority File (retrieved January 12, 2014), http://viaf.org