frontiers in Digital Humanities

ORIGINAL RESEARCH
published: 15 March 2017
doi: 10.3389/fdigh.2017.00005

Interlinking Large-scale Library Data with Authority Records

Felix Bensmann, Benjamin Zapilko and Philipp Mayr*

GESIS – Leibniz Institute for the Social Sciences, Knowledge Technologies for the Social Sciences, Cologne, Germany

In the area of Linked Open Data (LOD), meaningful and high-performance interlinking of different datasets has become an ongoing challenge. Necessary tasks are supported by established standards and software, e.g., for the transformation, storage, interlinking, and publication of data. Our use case Swissbib <https://www.swissbib.ch/> is a well-known provider for bibliographic data in Switzerland representing various libraries and library networks. In this article, a case study is presented from the project linked.swissbib.ch which focuses on the preparation and publication of the Swissbib data by means of LOD. Data available in Marc21 XML are extracted from the Swissbib system and transformed into an RDF/XML representation. From approximately 21 million monolithic records, the author information is extracted and interlinked with authority files from the Virtual International Authority File (VIAF) and DBpedia. The links are used to extract additional data from the counterpart corpora. Afterward, data are pushed into an Elasticsearch index to make the data accessible for other components. As a demonstrator, a search portal is developed which presents the additional data and the generated links to users. In addition to that, a REST interface is developed in order to enable also access by other applications. A main obstacle in this project is the amount of data and the necessity of day-to-day (partial) updates. In the current situation, the data in Swissbib and in the external corpora are too large to be processed by established linking tools. The arising memory footprint prevents the correct functioning of these tools. Also triple stores are unhandy by revealing a massive overhead for import and update operations. Hence, we have developed procedures for extracting and shaping the data into a more suitable form, e.g., data are reduced to the necessary properties and blocked. For this purpose, we used sorted N-Triples as an intermediate data format. This method proved to be very promising as our preliminary results show. Our approach could establish 30,773 links to DBpedia and 20,714 links to VIAF and both link sets show high precision values and could be generated in reasonable expenditures of time.

Keywords: library data, authority data, link validation, record linking, author names, RDF

OPEN ACCESS

Edited by:
Günter Mühlberger,
University of Innsbruck, Austria

Reviewed by:
Ioanna N. Koukouni,
Independent Researcher, Greece
Andreas Degkwitz,
Humboldt University of Berlin, Germany
Rudolf Mumenthaler,
University of Applied Sciences HTW Chur, Switzerland

****Correspondence:***
Felix Bensmann
felix.bensmann@gesis.org

Specialty section:
This article was submitted to Cultural Heritage Digitization, a section of the journal Frontiers in Digital Humanities

Received: *14 October 2016*
Accepted: *09 February 2017*
Published: *15 March 2017*

Citation:
Bensmann F, Zapilko B and Mayr P (2017) Interlinking Large-scale Library Data with Authority Records. Front. Digit. Humanit. 4:5. doi: 10.3389/fdigh.2017.00005

1. INTRODUCTION

Linked Open Data (LOD) have been an issue for several years now and organizations from all over the world are making their data available to the public by means of LOD. This issue has also come to certain importance within libraries (Pohl, 2010; Baker et al., 2011) and other cultural heritage

institutions (Mayr et al., 2016). In 2014, the LOD cloud[1] showed 570 interlinked corpora. Among them, 10% were publication-centric corpora.[2] The real number of freely accessible LOD-datasets may be assumed to be by far larger. Recently, Smith-Yoshimura published a paper on a survey where she analyzed project activities of 112 linked data projects (Smith-Yoshimura, 2016). Among the overall 90 participants, 33.3% in 2014 and 51.7% in 2015 are libraries. The analysis emphasizes some of the recurring issues library projects face.

The integration of conventional datasets as LOD into the semantic web represents a challenge for itself. Therefore, it is not uncommon that datasets in the semantic web are generated only once from existing datasets through various domains that had specific purposes. Updates are as rare as is the reuse of these datasets.

The metadata catalog Swissbib is a joint project of numerous libraries and library networks from Switzerland. The partners maintain their local collections and the changes are merged into the composite catalog. After that, the catalog is published and users can access it via a search portal. Currently, Swissbib uses conventional data formats which are not associated with the semantic web. In our project linked.swissbib.ch, we integrate classic bibliographic data as LOD into the semantic web and preserve maintainability as well as searchability. Thereby, we produce an additional advantage for Swissbib and our users. All described operations are performed on the data stock of Swissbib. The catalog contains metadata from publications, authors, and topics. In linked.swissbib.ch, we want to provide our users with the data on a daily basis. Therefore, the additional information of linked resources has to be indexed together with their original data in a search index. From that point, the whole processing has to be repeated periodically. Also, a new search portal for displaying the enriched data has to be developed. All procedures are, in general, use case specific and have to be prepared individually. Currently, Swissbib contains approximately 21 million descriptions of publications each one with authors and other entries. They are administered by a central repository system connected with participating libraries. The data can be exported to MarcXML files.

We see challenges in the creation of a common data model which is suitable as a knowledge graph and as a model for the search index. Also, the data model needs to be mapped to the existing format. In doing so, completely different paradigms have to be harmonized without losing any information. Additionally, a vocabulary and optionally an ontology should be chosen. Further tasks like the generation of URIs for newly created resources and the disambiguation of duplicate resources need to be addressed. Challenges are also seen in the processing of the large amount of data. This appears during the transformation from MarcXML to RDF and interlinking. First of all, we focus on linking the person data with the Virtual International Authority File (VIAF) and DBpedia. Apart from the size of Swissbib, VIAF or DBpedia problems are also caused by data quality. As a critical requirement, the overall workflow must not take longer than the update interval. A solution needs to be found to work with data differentials.

This article is structured as follows: first, we describe related work in reference to other linked library projects and existing approaches to individual challenges we faced. Afterward, we present the approach of linked.swissbib.ch with its overall system design and a longer digression to the person linking. This is followed by a section describing our process outcome before we eventually close with a discussion of our findings.

2. RELATED WORK

Recently, renowned libraries like the Library of Congress,[3] the German National Library,[4] or the British Library[5] (see, e.g., Smith-Yoshimura, 2016) started making their bibliographic and authority data accessible to the public by means of LOD. The objective is to enable users to discover and use data easier, and to extend their own data collections by interlinking data to other external, e.g., the same authors or publications in other collections. Thus, LOD is found to have a high potential for libraries (Byrne and Goddard, 2010; Hannemann and Kett, 2010).

The Linked Data Principles[6] (2006) (Bizer et al., 2009) and the 5 Star Open Data Scheme[7] (2010) suggested by Tim Berners-Lee as guidelines for data publishing practices, are actually applied by many LOD projects. In consequence, frequently used publishing forms comprise but are not limited to:

- SPARQL endpoints;
- HTTP servers to dereference HTTP URIs;
- Provided links to other LOD corpora;
- RDF dumps;
- Various forms of search access, web platforms, etc.

The work steps all these approaches have in common are in general the extraction of the data from a legacy system, URI assignment, transformation into RDF, and—if applicable—resource disambiguation. The generation of cross links is often done only when or once the links are stored together with the mass data and subsequently shipped. Usually, the RDF representation of the original data is stored in a separate repository next to the original data or it is generated on-the-fly. Rarely, the system is migrated completely to RDF.

2.1. Other Library Projects

From the vast number of projects that address LOD publishing a subset was chosen to be described here.

Haslhofer and Isaac (2011) and Isaac and Haslhofer (2013) describe **data.europeana.eu**.[8] It is the linked data prototype

[1] http://lod-cloud.net/.
[2] http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state/.
[3] http://id.loc.gov/.
[4] http://dnb.de/EN/lds.
[5] http://bnb.data.bl.uk/.
[6] https://www.w3.org/DesignIssues/LinkedData.html.
[7] http://5stardata.info/en/.
[8] http://data.europeana.eu.

of the Europeana Collections,[9] a digital library project that aggregates metadata about cultural items from more than 1,500 cultural institutions across Europe. It allows users to search and browse the metadata in a portal; the original cultural items can be accessed by browsing onto the item at its hosting institution. Europeana thereby acts as a common access point. data.europeana.eu serves a subset of these data as LOD. It offers data for more than 20 million cultural items like books, paintings, films, museum objects, and archival records. The Europeana Linked Open Data Pilot is uncoupled from the productive system. In undefined intervals, an RDF/XML-dump is created from the data basis and loaded into an RDF store. An HTTP server allows accessibility of the data in various formats determined through content negotiation. If an HTML representation is requested, the system redirects to the original Europeana service platform. Also, the data can be accessed via SPARQL or by downloading the dumps. Originally, the data are present in the Europeana Semantic Element (ESE) data model and are transformed into LOD within the Europeana Data Model (EDM) (Doerr et al., 2010), an RDF enabled data model, whose scope is to represent cultural items and to fit into the semantic web. The transformation takes place once with the whole target data by using an XSLT transformation. The necessary links to connect with the LOD web are provided from various Europeana related sources and connect, e.g., to Geonames (places), GEMET (topics), Semium ontology (time periods), and DBpedia (persons). data.europeana.eu also serves owl:sameAs links to already existing LOD corpora of partners.

data.bnf.fr[10] is a project of the French National Library targeting the provision of its collected data as LOD (Simon et al., 2013). The data are organized in various relational databases that include bibliographic records and more. In particular, there are 11 million bibliographic records from "Catalogue general," 150,000 documents from the Archives and Manuscripts database and 2,000,000 authority records of persons, organizations, works subjects, etc. The approach in this project consists of using the original databases for LOD publishing, also. A system based on the CubicWeb framework[11] queries the relational DBs and generates HTML and RDF data on-the-fly to be provided via an HTTP service. Once stored, each record type is assigned to an RDF class. According to the authors, CubicWeb can be adapted easily for further types. For linking, data.bnf.fr relies on already available alignments from previous projects that had to be resolved in order to use URIs. The strategy is to link primarily against LOD hubs like DBpedia and VIAF for better results. In total, 169,290 approximate and exact matches were generated with public corpora, among them DBpedia with 5,488 and VIAF with 15,937 links. Also, an alignment of bnf.fr FRBR manifestations with their respective FRBR works was carried out using a string-based approach. Additionally, they worked on a more sophisticated approach using machine learning techniques. The data.bnf.fr portal is operated in parallel to the known bnf.fr portal. CubicWeb offers a proprietary query language named Relations Query Language (RQL) that is similar to SPARQL; it also provides a functionality to translate a subset of SPARQL into RQL. Thus, a SPARQL endpoint could be offered. Links are dereferencable by the CubicWeb framework itself. Dumps of the data are provided in various RDF formats. A public search interface is also available.

2.2. Challenge Specific Approaches

Along with the common challenges of LOD publishing projects such as vocabulary mapping, transformation, or URI generation (cool URIs (cf. Sauermann et al., 2007)), library projects often focus on specific challenges (cf. Byrne and Goddard, 2010; Smith-Yoshimura, 2016), for example, classification, authority control and disambiguation, large-scale interlinking, maintaining large-scale data/links, high-performance access/search and even licensing.

For starting an approach, a model has to be developed that forms the foundation for the representation of the data as RDF. Such a model requires a vocabulary and an ontology which are suitable for publication and also are easy to use and to understand. The common consent is to reuse existing vocabularies where possible (Bizer et al., 2009; Schaible et al., 2016). Europeana, for example, developed the EDM to have a common subset to represent the involved datasets (Doerr et al., 2010), whereas data.bnf.fr uses an approach that provides a linked perspective of otherwise unchanged data. The perspective can be altered as necessary (Simon et al., 2013).

Depending on the general underlying system architecture, a transformation can happen on-the-fly for every query on a small amount of data, or once for the whole data, or, alternatively, a workaround could be found. Several case-specific transformation solutions seem to exist that cannot be applied in common scenarios. Though some tools focus on easing exactly that problem, e.g., Karma, a large data integration software suited for non-domain experts allows for a schema modeling and data transformation from and into various data formats (Knoblock et al., 2011). Another approach suitable for domain experts is Metafacture (MF) (Geipel et al., 2015) developed by the Deutsche Nationalbibliothek. MF requires good knowledge about the underlying data but can do transformations in a flexible and fast way. Yet another approach is the aforementioned CubicWeb which creates RDF resources on-the-fly; however, it requires software developer skills in order to be used.

Linking is required in order to interconnect with the LOD cloud and the semantic web in general. When registering with the LOD cloud, respectively the Data Hub,[12] it is even required to provide a certain amount of links. These links that point to pre-existing datasets in the cloud are a precondition to be listed officially in the LOD cloud. A link can be described by any RDF statement connecting two different resources, but its most common form is owl:sameAs. Common linking targets, the so called linking hubs, are e.g. the Virtual International Authority File

[9] http://www.europeana.eu/portal/de.
[10] http://data.bnf.fr.
[11] https://www.cubicweb.org/.
[12] https://datahub.io/.

(VIAF), DBpedia, or Geonames. VIAF provides authority data about persons and organizations collected from many national libraries; the corpus is also exposed as LOD. DBpedia provides data about various concepts extracted from Wikipedia and it covers person data. Geonames provides data about places. The authority files are especially interesting for libraries (Papadakis et al., 2015). Linking to these hubs is a recurring task mostly carried out by comparing the properties of two candidates, but it can also be done with various reasoning approaches. In Maali et al. (2011), the authors discuss various approaches to access the hubs and to link to them. Efficient linking has been an issue to many research activities and resulted in various tools, a short list of which can be found here.[13]

The Silk framework (Volz et al., 2009a,b) and LIMES (Ngomo and Auer, 2011) are examples for mature software tools that are ready for use and designed for interlinking RDF. However, they require the user to know the structure of the involved datasets and to define the linking process manually. OpenRefine,[14] formerly known as GoogleRefine, and its extension LODRefine[15] provide a GUI, so users are able to inspect and manipulate datasets. Another approach for software developers is the nazca[16] library, a python package for data alignment, also used by data.bnf.fr (Simon et al., 2013).

Since some datasets may be too big to be processed by common approaches in Gawriljuk et al. (2016), the authors present a matching process that deals with large-scale data by comparing hash values instead of whole resources. However, in the end, identifying corresponding resources is also a question of available reference material and corpus size.

3. METHODOLOGY

In this article, we present an approach to publish bibliographic mass data as LOD. We trace the path from legacy export over interlinking and enrichment to publication. Our methods include

- Schema migration using Metafacture.
- Data indexing with Elasticsearch.
- Implementation of a search portal and a REST interface.
- Data linking to large and heterogeneous corpora.
  - Data preparation using sorted lists of statements.
  - Blocking and parallel linking execution.
  - Data extraction and enrichment based on the sorted statements.

3.1. linked.swissbib.ch

The linked.swissbib.ch LOD platform aims to be an extension of the original Swissbib. Swissbib aggregates bibliographic records from multiple libraries and library networks in a Central Bibliographic System[17] (CBS), where data are merged and then indexed in SOLR search indexes. Users can access these data via the Swissbib web portal[18] by making use of the "Search" and "Browse" functionality.

Our transformation workflow sets in at this point. We use the CBS to create MarcXML dump files of the Swissbib corpus. We started by using Metafacture (MF) to convert the data into an RDF-based data model which has been created for linked.swissbib.ch. Thereby, the records are subdivided into different object types representing various bibliographic concepts. Resulting resources can be interlinked with external corpora individually or directly indexed into an Elasticsearch[19] (ES) index. Data from the index become available for publishing through an HTTP service and our comprehensive web portal. For the future, the next version of the portal will provide semantic search access and some newly introduced UI concepts. Meanwhile, the author names (persons), as opposed to organizational authors (organizations), are subject to interlinking and enrichment procedures. In a first step, persons are linked with VIAF and DBpedia, and in a second step the newly found links are added to the original person data. In a third step, we also retrieve the linked resources from the target corpus and incorporate the additional data. Eventually, the newly enriched persons are indexed as well to complete the linked corpus.

In the later demonstrator phase, the procedure has to be executed initially with the whole Swissbib dataset and after that it can be operated with incremental updates, unless external data changes. In that case, the enrichment has to be carried out once again using the new data.

Figure 1 depicts the overall workflow and system architecture. It shows the data export from the CBS as well as subsequent transformation and indexing steps. The double line represents the bibliographic records that are indexed directly, whereas the single line marks the persons' data way through the interlinking and enrichment procedures. The front-end components are shown accessing the search index (see right block in **Figure 1**).

Individual steps are introduced in more detail below.

3.2. Extraction, Transformation, and Presentation

The linked.swissbib.ch data model was designed to optimally represent our metadata. **Figure 2** shows a simplified model focusing on two concepts and their relations, namely dct:Bibliographic Resource and foaf:Person. BibliographicResource represents the bibliographic record, e.g., a book or an article of an author that manifests in various items; it stores a title, media type, subject, place of publication, and more. It also provides links to its contributors which can be persons or organizations. Persons have a first name, last name, birth year, death year, etc.

For data transformation, we use MF, a tool that implements a pipes-and-filters architecture to process data in a streamline fashion (Geipel et al., 2015). The pipeline can be individually arranged and extended with custom filters. First of all, the metadata is

[13] https://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/EquivalenceMining.
[14] http://openrefine.org/.
[15] https://github.com/sparkica/LODRefine.
[16] https://www.logilab.org/project/nazca.
[17] CBS is a metadata management system for libraries developed by OCLC.
[18] https://www.swissbib.ch/.
[19] https://www.elastic.co/products/elasticsearch.

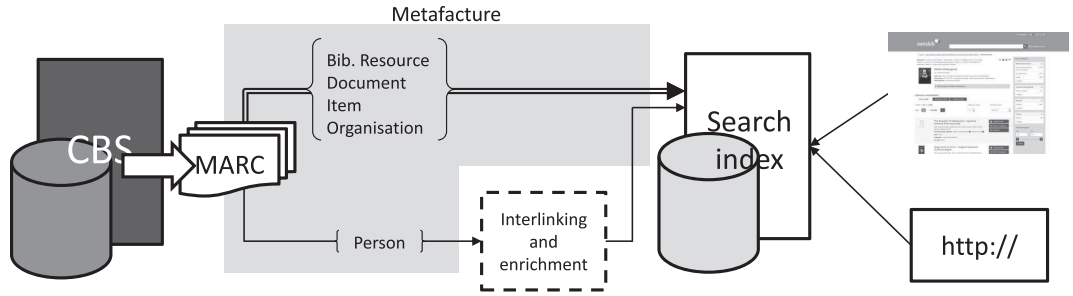

FIGURE 1 | **Actors and interactions in the linked.swissbib.ch workflow**.

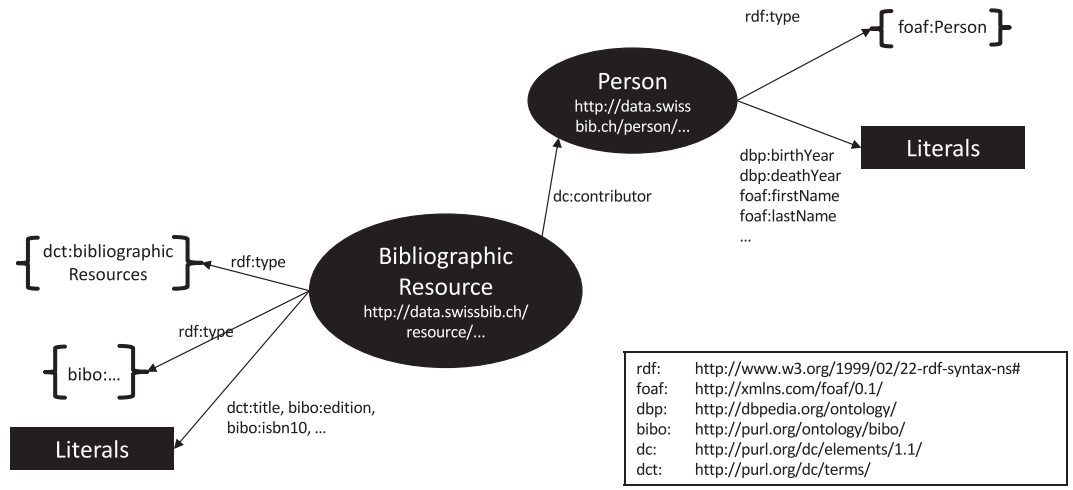

FIGURE 2 | **Linked.swissbib.ch data model**.

exported from the CBS into MarcXML files. These files are consumed with our MF arrangement and distributed in different pipes according to the concept they are assigned to. During MF processing, the data reside in a MF-specific data format.

The original data model is centric toward bibliographic records. This means that only bibliographic records are unambiguous. Authors, for example, are ambiguous. Thus, special procedures are used to assign persistent URIs to the resources. Where possible, these URIs are built from a unique data fragment of a resource, e.g., an ID. If a resource is exported from the CBS for a second time, we should ensure that it gets assigned with the same URI. This is required by the "Cool URI"-specification.[20] In cases where such a data fragment is not available, we risk to assign the same URI to different resources. Our solution is to calculate a hash value from specific literals of a resource (and directly referenced resources) that are unlikely to change or to appear more than once in that specific combination. This hash value becomes part of the URI. Our policy was to rather accept to produce ambiguous resources than to merge different ones, whereas merging identical resources is desired. If required, additional disambiguation steps can be executed afterward.

All resources except for the persons are transformed into ES JSON-LD bulk format[21] and subsequently indexed. Yet, the persons are stored to the local hard disk drive in standard JSON-LD files from where they are firstly interlinked, subsequently enriched and finally indexed with MF. The search index in use, as mentioned, is Elasticsearch, a well-known index based on Lucene as SOLR and is horizontally scalable across multiple hosts. It can efficiently serve multiple client requests and also index large datasets within a few hours. ES' dedication to JSON is most helpful in indexing the JSON-LD data, since it is a dialect of JSON and a lean RDF serialization. For each bibliographic concept, we created a type in the internal ES data model. When indexed, the RDF IDs and semantic relations are kept, and can be read out by the search portal and the REST interface. In the case of the search portal, ES also allows for a high-performance search. Nevertheless, apart from a good handling of JSON-LD, it is not designed for storing RDF, thus, lacking a SPARQL search. Also

[20] https://www.w3.org/TR/cooluris/.

[21] https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-bulk.html.

worth mentioning is that the necessarily applied schema limits the extendibility of the accustomed RDF model. In terms of user access, we allow for dereferencing the resources by providing a REST interface implemented with Hydra. Via this interface, we offer JSON-LD representations of our records.

As a further component, linked.swissbib.ch introduces a new search portal based on the look and feel of the "classic" Swissbib portal. This presentation component is the central starting point for users and offers search and browsing functionalities. It also extends the original platform adding the following new features.

- **Extended autocomplete for search terms:** this function suggests literature and other media, persons, and topics. Furthermore, it allows for a search in linked and enriched data that includes, i.e., alternative name spellings or pseudonyms.
- **Aggregated information sites:** these sites present detailed information about authors and topics. The information is collected from across various interconnected entities. Beyond this, recommendations for further research are made based on similarities between the entities. **Figure 3** shows a screenshot of such a site for a single author.
- **Knowledge cards:** these are pop-up windows with a brief description of the respective author or topic for orientation purposes.

The search portal was realized on the base of VuFind.[22] For accessing the ES index, we use advanced features like Multisearch to query multiple ES types at the same time, or partial loading of features. Using the search portal, users benefit directly from the linking, e.g., by pointing them to the respective resource at the original linking hub or other corpora, or by using the additional data for browsing.

3.3. Linking and Enrichment

The project linked.swissbib.ch realizes a contemporary and repeated linking of the actual data instead of just importing links from an external source. We want the corpus to be maintainable; Swissbib changes in content and size as well as the external corpora do. Therefore, linking procedures need to be flexible, fast, and able to rerun within short times. We have good knowledge about the structure of the Swissbib metadata (in MarcXML and LOD) but that does not apply to external linking candidates. A procedure needs to be adaptable for current and future linking candidates. Since we require the linking to be carried out in short time, we cannot rely on the availability of external services like the SPARQL endpoint from DBpedia, so, all data have to be locally available. As a consequence, we work with the RDF dumps offered by DBpedia and VIAF that we use not only to enrich our data with links but also to include some of the information we identify when linking to them. Additional data are stored and delivered along with the Swissbib metadata. Given the extensive amounts of data, this task is not trivial.

The interlinking procedure consists of three steps for every corpus to interlink: preprocessing, interlinking, and enrichment. In the described procedure, we process RDF on the level of statements as well as on the level of resources. By applying a preprocessing, we considerably reduce the effort for interlinking and enrichment. Thereby, we assume that the Swissbib corpus has a significant higher update rate than the external corpora (measured against the publication frequencies of new RDF dumps). This means we only need to preprocess the external corpora occasionally. In an operative mode, only Swissbib or a delta of it has to be preprocessed and then interlinking and enrichment can take place directly after that. The data flow diagram in **Figure 4** illustrates the procedure.

3.3.1. Preprocessing

Preprocessing collects the data files and converts them into the N-Triples format, thereby, we produce a long list of statements in a serialization form that can be stored on the hard disk drive. The statements can be read in a streaming-like manner to reduce memory consumption. In a second step, the statements are sorted alphabetically. Blank nodes are temporarily substituted by dummy-URIs. This ensures that all statements which describe a resource are stored cohesively, so we can process the data on the level of resources as well. It also means that resources are also ordered in sequence among each other. Later, this will help us align two sorted sets of statements/resources efficiently, namely $O(max(n, m))$ in the sorted case vs. $O(n*m)$ in the unsorted case, with n and m being the number of statements/resources in the two lists. **Figure 5** shows an example of person data in alphabetical order. Please note that the long forms of the URIs are considered for sorting.

Information unnecessary for the linking and enrichment is removed. Similarly, we build subsets that do not contain these data. We also remove duplicate statements which is easy to do, because the statements in question lie next to each other. Then, we extract the persons where necessary and subdivide them into blocks. **Figure 6** shows the principle. A certain block collects all resources that have pre-determined features in common which are relevant during the linking process. Given that we link by comparing first and last names and the birth year, we block by using the first letter of the last name of a person. Whenever an individual block exceeds a certain size we split it into several smaller blocks. As threshold we arbitrarily chose 200,000 statements. By applying a blocking we only have to link equivalent blocks instead of whole corpora.

3.3.2. Interlinking

For the linking, we rely on the aforementioned tool LIMES of the University of Leipzig. It provides a good performance and can be used from the command line. We describe the comparisons to be carried out by means of the domain-specific language. We achieve good results by comparing first names, last names, and birth dates requesting also full string matches. Using LIMES, it is possible to tune the linking for each case. A small Java-application is used to generate the configurations for each run. For this, it uses a template file and inserts the inputs and outputs for each pair of blocks.

The interlinking benefits from the controlled block sizes. In the past, we experienced problems using tools that stopped functioning properly at a certain size of input data. Having the

[22] http://vufind-org.github.io/vufind/.

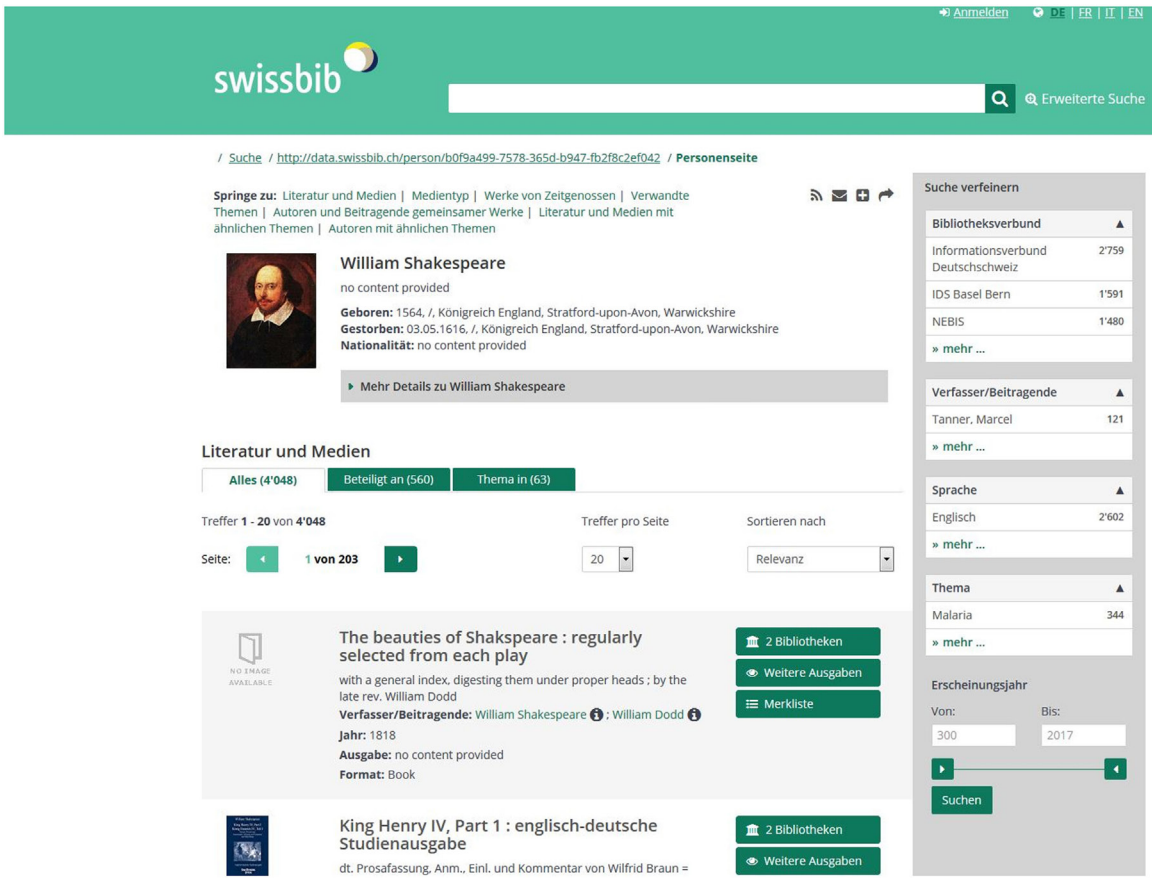

FIGURE 3 | Aggregated information page for the author "William Shakespeare".

configurations prepared, we can execute the interlinking in parallel, starting, e.g., 20 processes at a time. As a result, LIMES creates two files of owl:sameAs links for every linking process. One file with accepted links and one file with links for review. For the time being, we only use the accepted links.

3.3.3. Enrichment

For this step, we take the links <swissbib> <owl:sameAs> <viaf>, sort them by their object and align them with the respective external corpus. Thereby, we extract from the external corpus selected statements about the referenced author and rewrite them to make them statements of the Swissbib author resource. Particular statements of these refer to further resources instead of literals, e.g., locations. In order to be able to display these resources on the GUI, we summarize them in a single literal that represents the resource in a suitable manner. To do this, we use, e.g., labels or descriptions. The resulting literal is additional to the original property, added to the person description using a new extended property (dbp:birthPlace->swissbib:dbpBirthPlaceAsLiteral).

Finally, all persons, together with the links and extracted data, are deposited at an agreed location for indexing.

3.4. Linking Optimization

We have taken various steps to find the best linking configuration. Our metrics include processing time, number of links, and precision. Due to available data in Swissbib, we had foaf:name, foaf:firstName, foaf:lastName, dbp:birthYear, and dbp:deathYear as actual information carriers at our disposal. Further properties like skos:note or rdfs:type do not carry relevant information for interlinking. However, those exist in different extensions in the person resources. Available data (from August 2016) contained mostly persons with foaf:firstName and foaf:lastName (99.97%); less contained also dbp:birthYear (3.17%) and even less a dbp:deathYear (1.36%). Persons with foaf:name were extremely rare. In addition, there is bibliographic information of publications from the authors; however, these are again rarely present in the link targets.

On this base, we executed and compared the interlinking for first name–last name, first name–last name–birth year, and first name–last name–birth year–death year. As metric for evaluation, we had to rely on the precision value. A determination of the recall was not possible due to lacking ground truth. For the calculation of precision, we manually validated 100 links from each linking. The comparisons of first and last names as the only

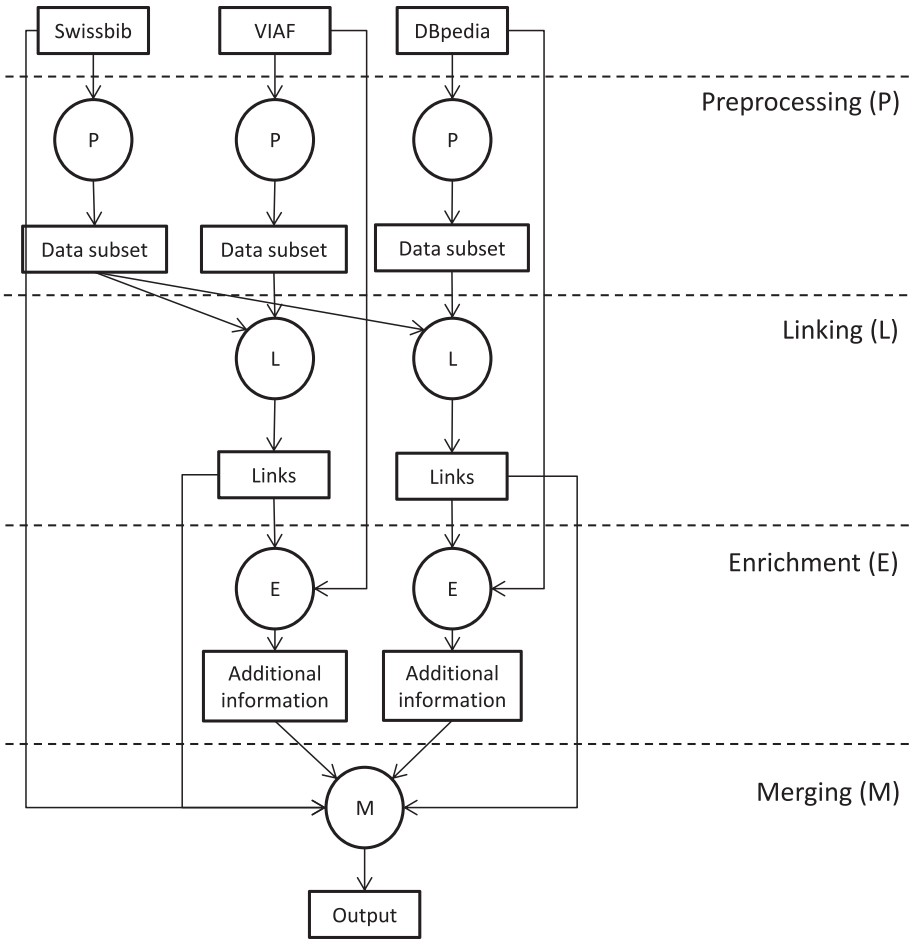

FIGURE 4 | **Data flow diagram of the interlinking procedure**.

```
<http://data.swissbib.ch/Person/a75e72e7e22d> <rdf:type> <http://xmlns.com/foaf/0.1/Person> .
<http://data.swissbib.ch/Person/a75e72e7e22d> <rdfs:label> "Wiltz, Marc" .
<http://data.swissbib.ch/Person/a75e72e7e22d> <rdfs:label> "Wiltz, marc" .
<http://data.swissbib.ch/Person/a75e72e7e22d> <foaf:firstName> "Marc" .
<http://data.swissbib.ch/Person/a75e72e7e22d> <foaf:firstName> "marc" .
<http://data.swissbib.ch/Person/a75e72e7e22d> <foaf:lastName> "Wiltz" .
<http://data.swissbib.ch/Person/b1c62f88c3c8> <rdf:type> <http://xmlns.com/foaf/0.1/Person> .
<http://data.swissbib.ch/Person/b1c62f88c3c8> <rdfs:label> "Gnauck, Hiltrud" .
<http://data.swissbib.ch/Person/b1c62f88c3c8> <foaf:firstName> "Hiltrud" .
<http://data.swissbib.ch/Person/b1c62f88c3c8> <foaf:lastName> "Gnauck" .
```

FIGURE 5 | **Person data as sorted N-Triples**.

criteria cause an insufficient precision. However, when we also included the birth year we achieved a sufficient precision that was no longer possible to improve further with the addition of the death year. At the same time, the number of links we found decreased greatly. We present the corresponding numbers in the following chapter. In the long run, we only link persons having first and last names and birth year. We are able to process the remaining persons together with the other resources.

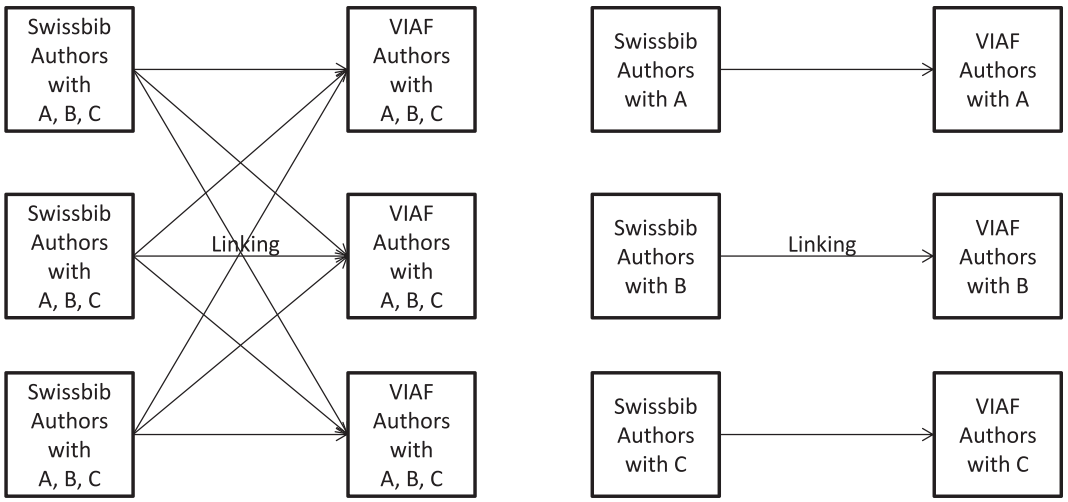

FIGURE 6 | **Interlinking without blocking (l) in comparison to blocking (r)**.

One measure to increase the link set could be the usage of person data from DBpedia that was omitted so far for reasons of complexity. As a consequence, the overall workflow will be delayed by the larger amount of data and the requirement of additional working steps. Currently, this looks feasible. We also see possibilities to generate more links by examining the links from the first name–last name comparison and use authors publications, if present, as new evidence to fine-tune the linking. In later iterations, further resource types could be interlinked. These could then be used in more complex link generation scenarios that involve more than one resource types. However, the implications for the resulting runtime behavior are not predictable.

4. RESULTS OF THE INTERLINKING WORKFLOW

In this section, we present the results from our workflow and the interlinking process. We report execution times, data composition, dataset sizes, and throughput. All tasks were conducted on a server in the Swissbib datacenter that is also intended to serve as productive system. The system has six Intel(R) Xeon(R) CPU E5-2660 v3 @ 2.60 GHz, 31 GiB RAM, and 500 GiB SSD. The baseline transformation and the interlinking and enrichment procedures were executed in sequence but may be parallelized later on.

4.1. Overall Workflow Performance

As mentioned above, the workflow is to a certain extent time-sensitive because we aim to provide the most up-to-date data. For an overview, we examined the time shares in the workflow.

Table 1 shows how long the workflow takes for its individual phases.

The two columns on the left hold the processing time for the baseline processing. The baseline consists of part of the MF pipeline that processes resources not intended for interlinking. Such resources, for example, may include bibliographic resources, documents, organizations, and items, and they represent the major part of data. The columns on the right hold the times for the other branch of the MF pipeline, the enrichment line. The columns also include linking and enrichment tasks and the indexing. However, the amount of data processed is limited to persons. It turned out that runtimes are suitable for our purposes. There are even reserves to further extend our workflows, e.g., to create enrichments for more resources, or intensify the existing enrichment procedures by adding further steps. Appending tests will have to show how our processes perform on slower hardware.

The individual runtimes for preprocessing, linking, enrichment and merging are presented in **Table 2**.

Given that the update frequencies of DBpedia and VIAF are small, we only have to run the preprocessing of both of them occasionally, whereas Swissbib has to be preprocessed every time. This eliminates the two largest time spans from the final duration, which is 3:30:52. Otherwise, the times comply with the other findings.

4.2. Composition and Sizes of Swissbib and the External Corpora

The data dump we use from Swissbib is from 2016-08-16 and contains the whole person data. In case of DBpedia, the data consist of individual datasets published by DBpedia, and we use the canonical as well as the localized datasets for the languages English, German, French, and Italian. The actual datasets are listed in the Appendix. For VIAF, we use the dump from 2016-08-12.[23] **Table 3** provides an overview of the statements which are subject to the preprocessing for linking.

[23] http://viaf.org/viaf/data/.

TABLE 1 | Time shares in the workflow (times roughly rounded).

Baseline	3.5 h	Enrichment line	1 h
		Preprocessing, linking, enrichment and merging	3.5 h
		Indexing	7 m
Sum	3.5 h	Sum	4,5 h

TABLE 2 | Time exposure for the different work steps.

	Swissbib	DBpedia	VIAF
Preprocessing	1:26:55	3:38:46	7:18:01
Linking	–	0:37:26	0:46:32
Enrichment	–	0:14:30	0:09:52
Merging		0:15:37	

TABLE 3 | Composition of corpora with statements relevant for linking.

	Swissbib	DBpedia	VIAF
#Total statements	21,982,204	138,193,546	664,121,467
#Persons	5,323,627	1,507,501	16,445,184
#Persons having first name, last name	5,322,140	1,117,102	5,404,363
#Persons having first name, last name, birth year	168,809	895,459	2,401,667
#Persons having first name, last name, birth year, death year	72,468	382,788	788,669

The row "#total statements" shows the total number of statements in the dataset used as input for the preprocessing (this is not the whole corpus). The row "#persons" reports the number of distinct resources that have foaf:Person, again foaf:Person, or schema:Person as rdf:type. The next three rows count the number of persons that exhibit the necessary properties used in the linking at least once. As already mentioned, these are first names, last names, and birth year. The person is also counted when properties appear more than once.

4.3. Link Validation

Link validation is performed via intellectual assessments. For this purpose, we implemented a tool called linkinspect[24] that allows human assessors to inspect and compare the two interlinked resources in a table view. The assessor then can mark the links as either correct, incorrect, or undecidable in cases where evidence for a clear decision lacks. For that reason, 100 random sample links are arbitrarily picked from one of the six link sets. These sets were created from linking Swissbib with each DBpedia and VIAF using the three combinations of properties presented above. The information about link source and target was made available in triplestores and is accessed by linkinspect via SPARQL. In Swissbib, the information consist mostly of the works an author has contributed to, but in DBpedia there is also a few general information about a person and sometimes descriptions of various works a person has contributed to. In VIAF, additional information is rare, though it is indeed possible to discover further information and works of a certain author by looking up its URL on the VIAF web page. Additional to the data provided by the triplestores, the validator is free to use further information sources.

Since we have not been able to assess how many links are correct in general we omit calculating the recall and concentrate just on precision. We conducted several runs with different choices for comparisons.

The results of the link validation are listed in the following **Tables 4** and **5**.

Table 4 shows the results of our validation of links to author names in DBpedia. Comparison of first and last names shows in the first column whereas in the second and the third column we added the birth and the death year, respectively. We can see that the precision (row: "Correct%") increases with the number of features in comparison. At the same time, the amount of links found decreases dramatically. The link outcome in the first name–last name row is significantly larger than the outcome of the other two rows, but only precision values of the other two are acceptable. So we decided to pursue the second variant for the final workflow.

In **Table 5**, we see a similar behavior and for this reason decided to use variant two (matching with "first name," "last name," and "birth year") for the same reasons. It is interesting to see that the amount of data of VIAF is by far larger than of DBpedia, whereas the amount of links found is significantly lower. Our explanation for this is a small overlap between Swissbib and VIAF. Also, we have to keep in mind that DBpedia consists of various editions of the same data in the localized datasets.

5. DISCUSSION AND FUTURE WORK

In this article, we present a system for the maintenance and publishing of bibliographic data as LOD. For this, we propose a regularly executed workflow based on existing Swissbib systems that provide users with refined data. In particular, we talk about an approach to flexibly deal with mass data and its interlinking that is currently not documented in similar projects.

By doing so, we rely on streaming-based processing with the tool Metafacture as well as subsequent indexing with Elasticsearch and a VuFind-based web frontend as the user interface. We addressed the challenge of interlinking mass data and heterogeneous corpora by applying an approach that is based on sorted lists of statements. The sorting keeps resources together and arranges them in alphabetical order. With this approach we thin out the data and optimize it for interlinking. By doing so it makes no difference which vocabulary is used in a corpus as long as the hierarchy is not too deep. On this basis, we were able to effortlessly realize a blocking that reduces the linking complexity and memory consumption. Also the sorting accelerates the alignment of links with the external data for the enrichment. Thereby our approach is explicitly suitable for extension with further processing steps.

5.1. Findings

Though by and large appropriate for its purpose, our workflow shows a few shortcomings that are also known from similar projects. We were partly able to solve them. In particular, we explain the difficulties in the following section.

[24]https://github.com/linked-swissbib/linkinspect.

TABLE 4 | Validation of the links to DBpedia.

	First name, last name	First name, last name, birth year	First name, last name, birth year, death year
Links	1,278,542	30,773	18,801
Samples	100	100	100
Correct %	29	93	85
Incorrect %	47	0	0
Undecidable %	24	7	15

TABLE 5 | Validation of the links to VIAF.

	First name, last name	First name, last name, birth year	First name, last name, birth year, death year
Links	4,371,727	20,714	5,317
Samples	100	100	100
Correct %	21	79	78
Incorrect %	34	1	0
Undecidable %	45	20	22

5.1.1. Ambiguous Authors

The Swissbib data are centric toward bibliographic works, not toward authors. This means that there is no authority control for authors. Authors can appear multiply. This is currently lessened by our way to generate the RDF-IDs, as was explained in section 3.2. However, a more sophisticated author disambiguation step would be a sensible measure. By using multiple language sets from DBpedia, we naturally link to ambiguous authors though we do not see a problem with this.

5.1.2. Procedural Issues in the Interlinking

The threshold used for the comparisons and the number and kind of properties included in these comparisons influence the quality of the linking. In section 4, we describe the link quality with respect to the properties chosen for the comparisons. Had we had more complete data, we would have been able to include more authors into the linking and compare more property values in order to improve the results.

5.1.3. Cost–Benefit Ratio

Currently, we perceive a rather small link outcome from the large datasets. This is not a problem resulting from the workflow or the linking itself but from the nature of the data and the overlap of the corpora. Since we do not know the amount of identical authors in the corpora, we cannot provide a value expressing this.

5.1.4. Tendencies

The data from Swissbib update once a day while the data from DBpedia depend on the cycles in which dumps are published, whereas VIAF lately increased its update frequency from biannual to rather monthly. In time when more datasets in DBpedia get canonicalized, we will perceive a down-shift in the number of linking results. In general, named corpora tend to grow, so with the increasing amount of authors in the datasets, effort for processing will increase over-proportionally; however, this appears to be rational for the foreseeable future.

5.1.5. Data Transformations

During its processing, the data are converted and represented in various different data models. This has to be done because particular preprocessing steps require the data in specific models. Each model has its advantages and disadvantages. However, some are more suited to be used in a particular technical process than others. The complete workflow comprises transformations from CBS to Marc21, from Marc21 to JSON-LD, and from JSON-LD to the Elasticsearch model. For the case that the person linking is included into the workflow, two additional transformations between two serializations of RDF are necessary: from JSON-LD to N-Triples and back. This is necessary in order to process the large amount of data in an efficient way. Despite this number of transformations, we can thereby ensure that no information in the data gets lost.

5.1.6. Manual Effort

The work on the linking in this project has shown that linking large-scale datasets still requires manual effort. Even if there are tools available for this task, a lot of preprocessing steps have to be executed in order that the data are in processible format and size. In most cases, these preprocessing steps cannot be done automatically since they are highly related to individual characteristics of the involved datasets.

5.1.7. Linked Data Publishing

Since Linked Data have a major role in this project, we have to be aware that eventually no Linked Open Data are published by means of the 5 star Linked Data paradigm. The generated Linked Data model is basis for the data imported into the Elasticsearch index. However, data are only searchable via this index including the link information which is generated through the underlying Linked Data model. Though all resources have their own URI which is dereferencable, Linked Open Data are not provided, e.g., via SPARQL as it would be required when being conformed to the 5 star Linked Data paradigm. Nevertheless, it is yet an open question whether an additional storage for the Linked Data serialization of Swissbib should be available just in order to fulfill these requirements.

5.2. Future Work

There are several aspects which can be addressed in the future.

One major part in this respect is to optimize the linking process itself. Preprocessing large-scale data for linking is still a time-consuming process. This process could be further optimized by using a pipes-and-filters architecture where data is passed between preprocessing steps (filters) without storing temporary results.

For linking persons from Swissbib with external data, currently only data which contain information on first and last name and birth year is used. In the future the number of links

between persons could be increased by also considering data with no information on the birth year. As a consequence, we have to be aware of the low precision of the generated links when only considering first names and last names for the interlinking.

In order to improve the results of interlinking persons, we could consider to also include data about their publications from German National Library or WorldCat.

A general challenge that could be addressed in the future is the author name disambiguation. Several approaches to face this challenge exist. In regard to literature data, an approach of using coauthor networks seems to be promising (see, e.g., Momeni and Mayr, 2016).

AUTHOR CONTRIBUTIONS

All authors listed have made substantial, direct, and intellectual contribution to the article and approved it for publication. FB wrote the main part, while BZ and PM provided considerable contributions.

ACKNOWLEDGMENTS

The authors thank René Schneider from Haute École de Gestion, Genève, Switzerland who included GESIS in the linked.swissbib.ch project. Individual works were carried out by their partners to whom the authors would like to express their special thanks. First, Günter Hipler and Sebastian Schüpbach from the University Library Basel for designing, setting up, and tuning the data conversion infrastructure. Also, Nicolas Prongué from Haute École de Gestion, Genève for conceptualizing, implementing, and validating the schema migration and finally Mara Hellstern, Lukas Toggenburger, and Philipp Kuntschik from the University of Applied Sciences HTW Chur who developed the web frontend.

FUNDING

The work described in this paper was funded by the project linked.swissbib.ch under grant no. 141-001 in the SUC P-2 program of Swiss Universities.

REFERENCES

Baker, T., Bermes, E., Coyle, K., Dunsire, G., Isaac, A., Murray, P., et al. (2011). Library linked data incubator group final report. Available from: https://www.w3.org/2005/Incubator/lld/XGR-lld-20111025/

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data-the story so far. *International Journal on Semantic Web and Information Systems* 5: 1–22. doi:10.4018/jswis.2009081901

Byrne, G., and Goddard, L. (2010). The strongest link: libraries and linked data. *D-Lib Magazine* 16: 1–11. doi:10.1045/november2010-byrne

Doerr, M., Gradmann, S., Hennicke, S., Isaac, A., Meghini, C., and van de Sompel, H. (2010). The Europeana data model (EDM). In *World Library and Information Congress: 76th IFLA General Conference and Assembly*. Gothenburg.

Gawriljuk, G., Harth, A., Knoblock, C.A., and Szekely, P. (2016). A scalable approach to incrementally building knowledge graphs. In *TPDL 2016 – 20th International Conference on Theory and Practice of Digital Libraries*, 1–12. Hannover.

Geipel, M.M., Böhme, C., and Hannemann, J. (2015). Metamorph: a transformation language for semi-structured data. *D-Lib Magazine* 21: 1. doi:10.1045/may2015-boehme

Hannemann, J., and Kett, J. (2010). Linked data for libraries. In *Proceedings of the World Library and Information Congress of the International Federation of Library Associations and Institutions (IFLA)*. Gothenburg.

Haslhofer, B., and Isaac, A. (2011). data.europeana.eu: the Europeana linked open data pilot. In *International Conference on Dublin Core and Metadata Applications*, 94–104. The Hague.

Isaac, A., and Haslhofer, B. (2013). Europeana linked open data – data.europeana.eu. *Semantic Web* 4: 291–7. doi:10.3233/SW-120092

Knoblock, C.A., Szekely, P., Ambite, J.L., Gupta, S., Goel, A., Muslea, M., et al. (2011). Interactively mapping data sources into the semantic web. In *Proceedings of the First International Conference on Linked Science*, Vol. 783, 13–24. Aachen, Germany: CEUR-WS.org.

Maali, F., Cyganiak, R., and Peristeras, V. (2011). Re-using cool URIs: entity reconciliation against LOD hubs. In *CEUR Workshop Proceedings*, 813. Hyderabad.

Mayr, P., Tudhope, D., Clarke, S.D., Zeng, M.L., and Lin, X. (2016). Recent applications of knowledge organization systems: introduction to a special issue. *International Journal of Digital Libraries* 17: 1–4. doi:10.1007/s00799-015-0167-x

Momeni, F., and Mayr, P. (2016). Evaluating co-authorship networks in author name disambiguation for common names. In *20th International Conference on Theory and Practice of Digital Libraries (TPDL 2016)*, 386–391. Hannover: Springer International Publishing.

Ngomo, A.-C.N., and Auer, S. (2011). Limes: a time-efficient approach for large-scale link discovery on the web of data. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, Vol. 3, 2312–2317. Barcelona: AAAI Press.

Papadakis, I., Kyprianos, K., and Stefanidakis, M. (2015). Linked data URIs and libraries: the story so far. *D-Lib Magazine* 21: 1–11. doi:10.1045/may2015-papadakis

Pohl, A. (2010). Linked Data und die Bibliothekswelt. Available from: http://eprints.rclis.org/15324/1/pohl_2011_linked-data_ODOK.pdf

Sauermann, L., Cyganiak, R., and Völkel, M. (2007). *Cool URIs for the Semantic Web*. Tech. Rep. Saarländische Universitäts- und Landesbibliothek, Kaiserslautern.

Schaible, J., Gottron, T., and Scherp, A. (2016). *TermPicker: Enabling the Reuse of Vocabulary Terms by Exploiting Data from the Linked Open Data Cloud*. Cham: Springer International Publishing. 101–17.

Simon, A., Wenz, R., Michel, V., and Di Mascio, A. (2013). Publishing bibliographic records on the web of data: opportunities for the BnF (French National Library). In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7882 LNCS (July)*, 563–577. Berlin.

Smith-Yoshimura, K. (2016). Analysis of international linked data survey for implementers. *D-Lib Magazine* 22: 1–10. doi:10.1045/july2016-smith-yoshimura

Volz, J., Bizer, C., Gaedke, M., and Kobilarov, G. (2009a). Discovering and maintaining links on the web of data. In *The Semantic Web – ISWC 2009*, Vol. 5823, 650–665. doi:10.1007/978-3-642-04930-9_41

Volz, J., Bizer, C., Gaedke, M., and Kobilarov, G. (2009b). Silk – a link discovery framework for the web of data. In *CEUR Workshop Proceedings*, Vol. 538. Madrid.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

APPENDIX

A. Composition of the Subset of the DBpedia Dump

The following two **Tables A1** and **A2** list the individual files from DBpedia that were used for the interlinking and enrichment process.

TABLE A1 | Data files from DBpedia used in the interlinking and enrichment process. Canonical datasets.

Description	URL
Mapping-based types—de	http://downloads.dbpedia.org/2015-04/core-i18n/de/instance-types-en-uris_de.nt.bz2
Mapping-based properties—de	http://downloads.dbpedia.org/2015-04/core-i18n/de/mappingbased-properties-en-uris_de.nt.bz2
Person data—de	http://downloads.dbpedia.org/2015-04/core-i18n/de/persondata-en-uris_de.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/de/long-abstracts-en-uris_de.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/de/images-en-uris_de.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/de/labels-en-uris_de.nt.bz2
Mapping-based types—en	http://downloads.dbpedia.org/2015-04/core-i18n/en/instance-types_en.nt.bz2
Mapping-based properties—en	http://downloads.dbpedia.org/2015-04/core-i18n/en/mappingbased-properties_en.nt.bz2
Person data—en	http://downloads.dbpedia.org/2015-04/core-i18n/en/persondata_en.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/en/long-abstracts_en.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/en/images_en.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/en/labels_en.nt.bz2
Mapping-based types—fr	http://downloads.dbpedia.org/2015-04/core-i18n/fr/instance-types-en-uris_fr.nt.bz2
Mapping-based properties—fr	http://downloads.dbpedia.org/2015-04/core-i18n/fr/mappingbased-properties-en-uris_fr.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/fr/long-abstracts-en-uris_fr.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/fr/images-en-uris_fr.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/fr/labels-en-uris_fr.nt.bz2
Mapping-based types—it	http://downloads.dbpedia.org/2015-04/core-i18n/it/instance-types-en-uris_it.nt.bz2
Mapping-based properties—it	http://downloads.dbpedia.org/2015-04/core-i18n/it/mappingbased-properties-en-uris_it.nt.bz2
Extended abstract	http://downloads.dbpedia.org/2015-04/core-i18n/it/long-abstracts-en-uris_it.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/it/images-en-uris_it.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/it/labels-en-uris_it.nt.bz2

TABLE A2 | Data files from DBpedia used in the interlinking and enrichment process. Localized datasets.

Description	URL
Mapping-based types—de	http://downloads.dbpedia.org/2015-04/core-i18n/de/instance-types_de.nt.bz2
Mapping-based properties—de	http://downloads.dbpedia.org/2015-04/core-i18n/de/mappingbased-properties_de.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/de/long-abstracts_de.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/de/images_de.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/de/labels_de.nt.bz2
Mapping-based types—fr	http://downloads.dbpedia.org/2015-04/core-i18n/fr/instance-types_fr.nt.bz2
Mapping-based properties—fr	http://downloads.dbpedia.org/2015-04/core-i18n/fr/mappingbased-properties_fr.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/fr/long-abstracts_fr.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/fr/images_fr.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/fr/labels_fr.nt.bz2
Mapping-based types—it	http://downloads.dbpedia.org/2015-04/core-i18n/it/instance-types_it.nt.bz2
Mapping-based properties—it	http://downloads.dbpedia.org/2015-04/core-i18n/it/mappingbased-properties_it.nt.bz2
Extended abstracts	http://downloads.dbpedia.org/2015-04/core-i18n/it/long-abstracts_it.nt.bz2
Images	http://downloads.dbpedia.org/2015-04/core-i18n/it/images_it.nt.bz2
Labels	http://downloads.dbpedia.org/2015-04/core-i18n/it/labels_it.nt.bz2