Zotero to Elexifinder: Collection, curation, and migration of bibliographical data

David Lindemann
david.lindemann@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

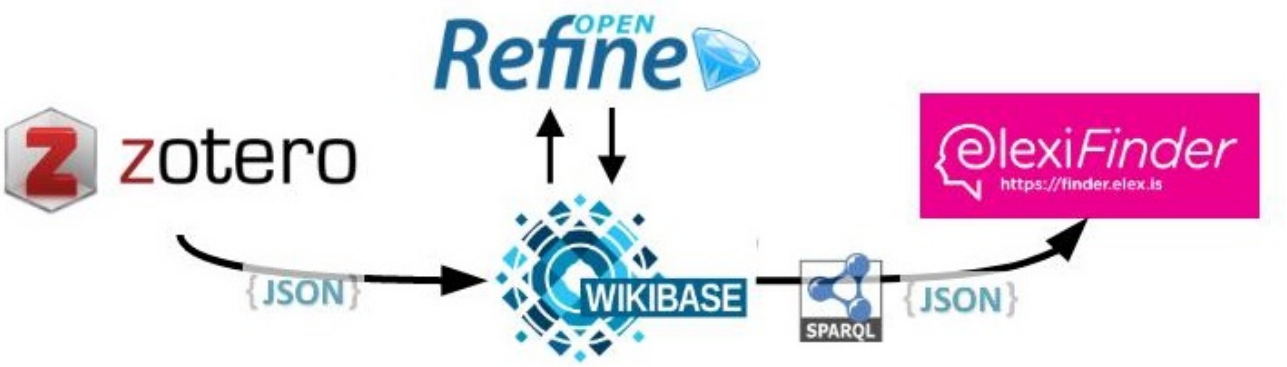

Figure 1: Zotero to Elexifinder workflow model

ABSTRACT

In this paper, we present ongoing work concerning a workflow and software tool pipeline for collecting and curating bibliographical data of the domain of Lexicography and Dictionary Research, and data export in a custom JSON format as required by the Elexifinder application, a discovery portal for lexicographic literature. We present the employed software tools, which are all freely available and open source. A Wikibase instance has been chosen as central data repository. We also present requirements for bibliographical data to be suitable for import into Elexifinder; these include disambiguation of entities like natural persons and natural languages, and a processing of article full texts. Beyond the domain of Lexicography, the described workflow is applicable in general to single-domain small scale digital bibliographies.

KEYWORDS

bibliographical data, author disambiguation, e-science corpora

1 INTRODUCTION

In 2019, version 1 of Elexifinder,[1] a discovery portal for lexicographic literature, was launched in the framework of the ELEXIS project [2].[2] At the same time, at University of Hildesheim, a domain ontology and bibliographical data collection for Lexicography and Dictionary Research was planned [6, 5]. Both endeavours already had compiled significant datasets. At a dedicated workshop connected to the 2019 eLex conference in Sintra (Portugal), it was decided to combine the efforts, and the workflow explained in this paper was designed, in order to merge existing datasets, decide criteria for data curation, and make the results available to the lexicographic community. Two years later, at the 2021 Euralex conference, Elexifinder version 2 was introduced [3]. Main shortcomings of Elexifinder version 1 have been sorted out, namely the missing author disambiguation, and the coverage of the domain's literature has been significantly increased, also regarding publication languages other than English. Moreover, a vocabulary of lexicographic terms has been developed, which is now used for content-describing indexation of article full texts.

Lexicography and Dictionary Research is a relatively small discipline, having thematic intersections with Corpus Linguistics, Terminology, Natural Language Processing, and Philology. In metalexicographic literature, all aspects of the lexicographic process, dictionary structure and functions, dictionary use, and other relevant issues are discussed. The lexicographic community communication is mainly taking place through a reduced number of conference series and journals, being complemented by handbooks and other edited volumes. The need for a dedicated digital bibliography arises from the following observations:

- The vast majority of publications do not have Digital Object Identifiers (DOI), and thus are not indexed in cross-domain digital collections of publication metadata. This applies to nearly all older publications, but also to many newer contributions published in the last two decades.
- When searching for metalexicographical publications in cross-domain digital collections, search results are mixed up with publications from other domains, which may disturb a straightforward information retrieval.
- Author disambiguation in domain-independent digital collections that can be considered the big players in the field (such as Google Scholar) is not at all accurate, so that very

[1] Accessible at https://finder.elex.is.
[2] See https://elex.is.

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

often name variants are not resolved to a single person entity, and different persons with the same name are not disambiguated.
- If articles are indexed with content-describing terms in cross-domain digital collections, the vast majority of those terms will be out of the scope of the domain we are looking at.
- Publication metadata found at big (i.e. automatically compiled) repositories is often incomplete or noisy, so that using those, e.g. for citations, requires manual intervention in order to achieve a publishable quality.

Therefore, it seems useful to provide the lexicographic community with a platform that makes publications and their metadata accessible in a way that the described shortcomings will be overcome. Single-domain endeavours of this kind, which all involve manual curation, are *DBLP*[3] for Computer Science, *IxTheo*[4] for Theology, or *EconBiz*[5] for Economics. Inspired by features found in these, we propose a workflow that involves the use of free software accessible to anybody, which makes it reproducible and cost-reducing.

2 LEXBIB ZOTERO GROUP

Zotero,[6] developed and maintained by the Corporation for Digital Scholarship[7], a non-profit organisation, is the most widely used open source citation management software application. Zotero offers functionality for web-scraping publication metadata, importing metadata from different structured formats, and an online platform for collaborative curation of metadata, along with the possibility to attach full text PDF (and TXT versions) to metadata records. The Zotero scraper functionality allows to download publication metadata and attached PDF files from all those sites the Zotero community has provided a "translator"[8] for, including the web platforms of major publishing houses, Open Journal Systems, etc. From the Zotero platform, users are able to obtain metadata records as single items or as batches for import into their own citation managers, or as export records in a range of citation styles or in structured formats such as bibtex. Members of a Zotero group can view and download full text attachments. Moreover, Zotero items can be annotated with custom tags, and additional information (such as excerpts or comments) can be attached to them. Around Zotero, an active community is developing plug-ins that add new functionalities to Zotero.[9]

In the first planning period of the LexBib project, funded by the University of Hildesheim, conference publications of the Euralex and the eLex conference series, and publications from a range of journals and edited volumes have been added to LexBib Zotero group.[10] Items collected for Elexifinder version 1, available as tabular data, have then been merged to the Zotero group. For this purpose, tabular csv data has been transformed to RIS format[11] and imported to Zotero. Additionally, metadata records from OBELEX-meta and EURALEX-Dykstra bibliographies have been added.[12] Duplicate management has been done in batches (whole journal issues or conference iterations), or one by one using Zotero's built-in duplicate detection functionality. Main criterion for the inclusion of metadata records has been the availability of the corresponding full texts. This means a clear preference for Open Access publications; but also other publications have been included, wherever a suitable license agreement allowed access to the text.[13]

Zotero data can be accessed by API,[14] or exported locally using pre-set or custom export scripts. We use an adapted version of the Zotero JSON-CSL exporter, which produces a list of JSON objects containing all metadata fields and their values as literal strings, as well as the location of all local file attachment copies. For statements that cannot be expressed using standard Zotero fields[15], we have used Zotero tags as workaround, following a simple syntax of predicate and object. For example, for asserting that an article is a review article, the tag ":type Review", and so on. Tags in Zotero can be easily copied from one item to others by manual drag-and-drop operations, set via API, and also be included in display styles, so that in the Zotero item listings, for example, review article titles can be preceded by a coloured symbol. With this workaround we can assert semantic triples inside Zotero. That is, for instance, that for representing the statement that a certain item is contained in another item (e.g. a book chapter item in an item of type book), we use a tag beginning with ":container", followed by an identifier for the containing item; for a conference paper presented at a certain event, we use a tag beginning with ":event", followed by an identifier for that event. For both of these, corresponding Zotero fields do exist ("contained in", "presented at"), but these are filled by the web scraping and importer translators with literal string values as needed for citations, and not with unambiguous identifiers.

For Elexifinder, a special metadatum is included in all publication metadata sets: The location of the first author. This allows the generation of location maps and search filters according to locations in the Elexifinder portal. For these locations, we insert English Wikipedia page titles in the Zotero "extra" field.[16]

3 LEXBIB WIKIBASE

3.1 Wikibase as LOD infrastructure solution

The decisive shift from a metadata set as in Zotero, which consists of certain fields and their literal values, towards unambiguous Linked Data lies in the reconciliation of those literal values against existing or new unambiguous identifiers. For example, and this already refers to the hardest nut to crack in this context, an author may have several name variants appearing across the publication metadata collection, and there may be other persons sharing the same name, or any of the name variants. But one author or editor (i.e., a "creator") should only have one identifier (such as ORCID). Since we do not know Wikidata and/or ORCID identifiers of all creators in our database, we need to create our own (and map them later). Other Zotero fields that should be

[3] Accessible at https://dblp.org/.
[4] Accessible at https://ixtheo.de/.
[5] Accessible at https://www.econbiz.de/.
[6] See https://zotero.org.
[7] See https://digitalscholar.org/.
[8] See https://www.zotero.org/support/translators.
[9] For example, very recently the Cita plug-in has been developed, which allows to add citation metadata to Zotero records, see https://meta.m.wikimedia.org/wiki/Wikicite/grant/WikiCite_addon_for_Zotero_with_citation_graph_support.
[10] Last version accessible at https://www.zotero.org/groups/lexbib/library.
[11] See https://en.wikipedia.org/wiki/RIS_(file_format).
[12] See references in [3].
[13] Article full text are stored and exclusively used for project-related text mining tasks; they cannot be downloaded from Zotero. We instead provide download links which lead to the download offered by the corresponding publisher, subject to applicable restrictions.
[14] See https://www.zotero.org/support/dev/web_api/v3/start.
[15] See https://www.zotero.org/support/kb/item_types_and_fields.
[16] Wikipedia page titles are unambiguous (see e.g. https://en.wikipedia.org/wiki/Cambridge vs. https://en.wikipedia.org/wiki/Cambridge,_Massachusetts), and map to only one Wikidata entity. This strategy has turned out effective, since manual annotators are able to find the adequate Wikipedia page without hassle.

reconciled against unambiguous identifiers are those describing the containing item, the conference where the contribution was presented, the journal, the publisher, the publication place, and the publication language. For some of these, persistent identifiers are available in many cases (e.g. journals), or in all cases (languages). In general, we create our own identifiers, and map them to Wikidata; in some cases, immediately (languages, places, and, by ISSN, also journals), and in other cases, we leave that mapping to the (near) future, as it is the case for creators and publishers. Other Zotero fields contain identifiers (ISSN, ISBN, DOI), which after normalisation can be taken directly as external identifiers in a Linked Database.

After experimenting with different RDF database solutions, which allow to represent data in the described way, we have decided for Wikibase,[17] which is the software infrastructure underlying main Wikidata.[18] Since 2019, "Wikibase as a Service" is offered to the community.[19] Wikibase entities are items (each of which has its own identifier preceded by the letter Q), and properties (preceded by letter P), just as in Wikidata, but in a different namespace. Properties may point to other items, other properties, external identifiers, or values of a certain datatype, such as "monolingual text", "point in time", "string", "url", etc.[20]

Wikibase as central data repository solution has several advantages compared to other infrastructure solutions for Linked Open Data (LOD):

- Entity data is displayed on entity pages, where it can be viewed and edited. These pages always reflect the last update.
- A complete edit history is available, and changes can be undone.
- Every entity page is linked to a dedicated discussion page.
- User and user rights management allow a community-driven editing process.
- In addition to query interface and SPARQL endpoint known from other RDF database solutions, Wikibase data can be uploaded and downloaded using an API, and as entity data dump in several formats.

The backbone of LexBib Wikibase is an ontology of classes and properties,[21] which can be aligned to Wikidata or other external ontologies. We have started to define these alignments. This ensures interoperability with other resources, such as Wikidata, so that data can be transferred from LexBib to Wikidata or vice versa, or accessed in both at the same time, using federated SPARQL queries.

3.2 Zotero to Wikibase migration

As mentioned before, Zotero item data is exported from a local Zotero instance, using an adapted version of the Zotero JSON-CSL exporter.[22] The resulting list of JSON objects is then processed in the following way:

- Zotero tags that contain semantic triple shortcodes (explained above) are mapped to the corresponding LexBib wikibase properties, in this case with datatype "item", that is, to object properties.
- Creator name and publisher name literals are mapped to the properties corresponding to the creator role (author or editor), or to the publisher. This is done in a way that the name literals appear as qualifiers to a wikibase "novalue" statement, which is a placeholder for the creator or publisher item, that will be defined in the disambiguation process explained below.
- Zotero fields that contain external identifiers (ISSN, ISBN and DOI), are mapped to the corresponding properties of datatype "external identifier". Wikibase properties of that datatype allow to define a URL pattern, in order to make the identifier a valid hyperlink, which can be clicked on in Wikibase entity data pages.
- As mentioned, we use the Zotero "extra" field ("note" in bibtex) for annotation of the item with a Wikipedia page that corresponds to the first author's location. Wikidata API is queried for the corresponding Wikidata entity, an equivalent of which is created in LexBib Wikibase, in order to function as object to the property "first author location".
- The Zotero "language" field, in LexBib may contain a two-letter ISO-639-1, or a three-letter ISO-639-3 code. This is mapped to a property pointing to the language item corresponding to that code.
- The Zotero item URI is taken as external identifier in LexBib wikibase, with the Zotero storage location of PDF and TXT attachments as qualifiers to that statement. In addition, we annotate this statement with a qualifier asserting the presence of an abstract, and, if any, in what language.[23]
- The content of the remaining fields is mapped to Wikibase properties of the corresponding datatype ("URL", "string", or "point in time").

The resulting dataset is then imported into LexBib Wikibase. It is worth mentioning that uploading data to a Wikibase triple by triple using the mediawiki API of the Wikibase instance[24] takes about 0.5 seconds per triple, which is due to the need of updating Wikibase search indices and edit histories for every single uploaded triple.

3.3 Entity disambiguation using Open Refine

The around 5,000 creator names appearing in LexBib Zotero by spring 2021 have been mapped to around 4,000 unique person items. This has been done testing different clustering algorithms available in the Open Refine application,[25] by Christiane Klaes from the University of Hildesheim, in the framework of her MA thesis [1]. These are the creator items present in LexBib Wikibase experimental version 2.[26]

From that moment on, any new Zotero item that is exported to Wikibase, which will contain, as explained above, one or more creator statements of type "novalue", is reconciled against existing LexBib Wikibase creator items, using the given and last name literal qualifiers. For this purpose, a reconciliation service for LexBib Wikibase is set up[27], and then accessed by Open Refine, in order to match creator name literals to creator items.

[17] See http://wikiba.se; our instance is accessible at http://lexbib.elex.is.
[18] Accessible at http://www.wikidata.org.
[19] See https://www.wbstack.com. The service has been co-enabled by Adam Shoreland (https://addshore.com/), Rhizome (https://rhizome.org/), and WMDE (https://www.wikimedia.de/).
[20] See https://www.wikidata.org/wiki/Help:Data_type.
[21] For more information, see LexBib Wikibase main page at https://lexbib.elex.is.
[22] Available at https://github.com/elexis-eu/elexifinder/blob/master/Zotero/LexBib_JSON.js.
[23] The abstract language is assumed to be the same as the publicaton language, if not stated different as tag shortcode ":abstractLang".
[24] For LexBib Wikibase, see https://lexbib.elex.is/w/api.php.
[25] Available at https://openrefine.org/.
[26] Accessible at https://data.lexbib.org.
[27] This is done using https://github.com/wetneb/openrefine-wikibase.

If a literal can not be matched to any existing item, a new person item is created. The reconciliation also works with fuzzy matches, and all name variants attached to existing items are considered. Matches can also be manually chosen. Any additional name variant appearing in Zotero data is linked to the LexBib Wikibase person item as "alias" label, while the most frequent name variant is chosen as "preferred" label. This allows for the new name variants being available for subsequent reconciliation iterations.

LexBib persons have up to six name variants found in Zotero data. In some cases, we have chosen the preferred name variant manually, according to the author's own choice, or to conventions in the community regarding the naming of commonly known authors.[28]

3.4 Full text processing

LexBib full text PDFs are stored in the local Zotero storage folder, which is automatically synchronised with Zotero cloud. When processing Zotero JSON output, PDF files are sent to an installation of the GROBID application[29], which will propose a TEI representation of the PDF content. This allows for isolating the full text body from the other text components, such as title, running titles, abstract, author list, and references section. The extracted full text body is manually validated, and, in case of any mistake, it is corrected, using a plain TXT version of the PDF, which is by default produced by Zotero.

GROBID turns out to structure PDF content as TEI very efficiently if the article resembles a typical structure as found in journals and proceedings. Book chapters and review articles, which normally do not feature an abstract, in turn, are usually not parsed adequately. In those cases, we now use directly the plain TXT version for producing a cleaned version manually.

The article text is then lemmatised,[30] and lexicalisations of LexVoc lexicographic terms are looked up in the text.[31] LexVoc vocabulary[32] is a resource still under development; for the term discovery process, terms and lexicalisations (labels) are obtained from LexBib Wikibase by a SPARQL query, the result of which will reflect the state of LexVoc in that particular moment. The keyword processor returns counts of every term, so that relative frequencies can be calculated for every term, according to the occurrences of its labels and the amount of tokens in the article text body; this information can be uploaded to LexBib Wikibase bibliographical items, so that term indexation becomes part of their entity data.

4 WIKIBASE TO ELEXIFINDER

The described workflow is necessary for being able to export bibliographical data in a custom JSON format, as needed for Elexifinder, which is an application based on some of the elements of the Event Registry system architecture [4]. In particular, authors and content-describing terms (Elexifinder "categories") have to be represented as objects containing an unambiguous URI and a textual label; the containing item, the LexBib Zotero item URI, and the link for accessing full text download are represented as URL, publication date in ISO 8601 format, publication language in ISO 639-3 format, and the item title as simple string.

The full text body itself is also exported to Elexifinder, where it is used for displaying the first bits of it in search result displays, and for wikification, from which Elexifinder "concepts" are obtained, as long as the system is able to associate named entities occurring in the text with Wikipedia pages that describe them.

5 CONCLUSIONS AND OUTLOOK

The described workflow enables us to disambiguate entities found in bibliographical datasets. For the time being, we are applying this for feeding the Elexifinder app. Having chosen Wikibase as central data repository also allows for aligning LexBib data with Wikidata in a straightforward way. In some cases, we have imported statements from Wikidata, in order to enrich LexBib entities with additional information, but that can be done the other way round as well. In other words: Wherever we find (or create) a Wikidata entity to align with our own, we can export the statements asserted on LexBib Wikibase to the main Wikidata. We have done this using LexBib events (conferences) as test case, and plan to align other entity types with Wikidata in the near future, namely articles, persons, and organisations.

ACKNOWLEDGMENTS

The research received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 731015.

REFERENCES

[1] Christiane Klaes. 2021. *Linked Open Data-Strategien zum Identity Management in einer Fachontologie*. Master's thesis. Universität Hildesheim, Hildesheim, (June 2021). http://lexbib.elex.is/entity/Q15468.

[2] Iztok Kosem and Simon Krek. 2019. ELEXIFINDER: A Tool for Searching Lexicographic Scientific Output. In *Electronic Lexicography in the 21st Century: Proceedings of the eLex 2019 Conference*. Lexical Computing CZ s.r.o., Brno, 506–518. http://lexbib.elex.is/entity/Q9484.

[3] Iztok Kosem and David Lindemann. 2021. New developments in Elexifinder, a discovery portal for lexicographic literature. In *Lexicography for Inclusion: Proceedings of the 19th EURALEX International Congress, 7-11 September 2021, Alexandroupolis, Vol. 2*. Democritus University of Thrace, Alexandroupolis, 759–766. http://lexbib.elex.is/entity/Q15467.

[4] Gregor Leban, Blaz Fortuna, Janez Brank, and Marko Grobelnik. 2014. Event registry: learning about world events from news. In *Proceedings of the 23rd International World Wide Web Conference, WWW14, Seoul, Korea, April 7-11, 2014*, 107–110. DOI: 10.1145/2567948.2577024.

[5] David Lindemann, Christiane Klaes, and Philipp Zumstein. 2019. Metalexicography as Knowledge Graph. *OASICS*, 70. http://lexbib.elex.is/entity/Q13955.

[6] David Lindemann, Fritz Kliche, and Ulrich Heid. 2018. LexBib: A Corpus and Bibliography of Metalexicographical Publications. In *Lexicography in Global Contexts: Proceedings of the 18th EURALEX International Congress, 17-21 July 2018, Ljubljana*. Ljubljana University Press, Ljubljana, 699–712. http://lexbib.elex.is/entity/Q6059.

[28] See an example at http://lexbib.elex.is/entity/Q1583.
[29] See https://grobid.readthedocs.io.
[30] For the time being, we are only processing English text. For lemmatisation, we use spaCy (see https://spacy.io/).
[31] This is done using https://pypi.org/project/flashtext/.
[32] Described at http://lexbib.elex.is/wiki/LexVoc.