Collaborative Literature Work in the Research Publication Process: The Cogeneration of Citation Networks as Example

Leon Otto Burkard and Andreas Geyer-Schulz

Abstract In educational and scientific publishing processes scientists and prospective scientists (students) in their different roles (author, editor, reviewer, production editor, lector, reference librarians) invest a large amount of work into the proper handling of scientific literature in the widest sense. In this contribution we introduce the LitObject middleware and its combination with the popular open-source tool Zotero. The LitObject middleware supports the exchange of sets of scientific objects (literature objects) consisting of bibliographic references and documents (e.g. PDF-documents) by scientists. In our contribution we emphasize several process improvements with a special focus on the cogeneration of citation networks.

1 Introduction

In every publication literature is cited. To be able to cite literature the authors of a publication have to search, find, possibly evaluate and read the cited publications, file the fulltext document and, later, have to retrieve the publications filed. This time-consuming process of literature work has to be done by every researcher. To facilitate literature management scientists can use literature management software such as EndNote, Mendeley or Zotero that are compared and described in more detail by Hensley (2011). Based on the list of expectations from Gilmour and Cobus-Kuo (2011), management of literature also includes, for example, the import of references and fulltext documents from digital libraries, gathering metadata from documents as well as the organization in a database and annotation of literature. Additionally, properly formatted citations should be provided in various styles by the software. Literature management tools support the work with literature objects by providing functions to arrange them logically, for example, in a hierarchic folder structure, linking literature objects and in most cases provide a fulltext search for all organized literature objects and options to annotate them. In this paper the bundle

L.O. Burkard (✉) • A. Geyer-Schulz
Information Services and Electronic Markets, Karlsruhe Institute of Technology, Karlsruhe, Germany
e-mail: mail@leon-burkard.de; andreas.geyer-schulz@kit.edu

A.F.X. Wilhelm, H.A. Kestler (eds.), *Analysis of Large and Complex Data*, Studies in Classification, Data Analysis, and Knowledge Organization,
DOI 10.1007/978-3-319-25226-1_52

of reference and fulltext documents will be called a *literature object*. Literature objects are managed with the help of a PDF-manager as introduced by Mead and Berryman (2010). A PDF-manager is a software application that manages not only a reference but also mainly fulltext documents in PDF-format. Unfortunately, as described by Hull et al. (2008), metadata support for retrieving correct references into PDF-manager software is error-prone: There is no "universal method to retrieve metadata. For any given publication, it is not possible for a machine or human to retrieve metadata using a standard method" (Hull et al. 2008, p. 8). Also there are various options how metadata can be represented. The PDF-format itself only offers insufficient and limited metadata fields to embed metadata and as a consequence this feature is used only rarely according to Howison and Goodrum (2004).

Apart from new ways of organizing and managing literature the way of publishing changes: Glänzel and Schubert (2004) examined that in the 1980s about 25 % of all publications had only one author. This percentage decreased to 11 % until the year 2000. The average journal publication in 2011 had more than four authors. In the field of computer science single author publications represented only about 15 %, the majority was written by three and more authors according to Solomon (2009). However, the most common used literature management tools operate as user desktop applications with proprietary and restricted sharing and collaboration features. Even worse is the situation on the tool support side for extended use-cases with requirements such as the circulation of literature within the scope of fair use. One application for this requirement could be the temporary access to the authors' cited literature by the lector, reviewer or editor of the publication within the review process. Another use case is the support of collaborative literature work to facilitate the activity of writing a multi-author publication.

In order to improve the part of collaborative literature work we developed the LitObject middleware. The LitObject middleware serves as a foundation for various extended services that require a structured access to literature objects. As an example for an extended service we present the utilization of the LitObject middleware as a cogenerated data basis for citation networks.

2 A Simplified Publication Process

For a better motivation and understanding of the necessity for a middleware for literature objects we introduce a simplified publication process as depicted in Fig. 1. A detailed conceptual description of the publication process with an emphasis on the author's and editor's tasks can be found in University of Chicago Press (1982). The high-level perspective process consists of four main subprocesses:

In the first subprocess *Creation* the article for the first submission is prepared. This subprocess includes literature work, particularly retrieving literature objects and using references in the written document. After submission of the article the subprocess *scientific quality management (SQM)* starts. *SQM* includes several subprocesses such as the whole review process including the prior selection of

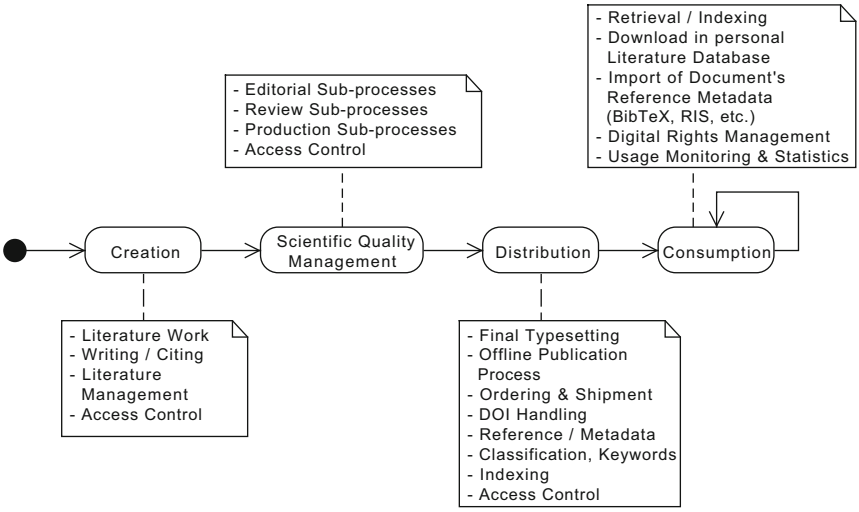

Fig. 1 A simplified publication process

appropriate reviewers as well as returning the submission to the author(s) for improvement, correction and submission of a camera ready version, editorial decisions about the orientation and corresponding selection of submissions and, last but not least, all tasks of a production editor as key person for the timely completion and coordination of the volume. The subprocess *Distribution* concentrates on the technical publication steps starting with the final typesetting of the submission for print and online publication followed by tasks for the correct classification, indexing as well as DOI handling and generation, respectively, as well as provision of reference metadata for usage in libraries, online catalogs, search engines and literature management environments. Also traditional ordering and shipment processes in combination with logistic, payment and accounting services are affected. Tasks in the *Distribution* subprocess are commonly executed by personnel in publishing companies, *SQM* usually by academic volunteers and writing articles in the subprocess *Creation* by authors. Finally, in the *Consumption* subprocess the corresponding publication is found, retrieved, (delivered and payed), read, filed and cited. This subprocess also includes usage monitoring as feedback for authors and the editorial board.

2.1 Challenges in the Publication Process with Regard to Literature

Challenges in the *Creation* subprocess are: How can past literature searches be rediscovered? How can literature objects be shared in multi-author publication

scenarios and how, in general, can the result of literature searches executed by scientists or students be stored permanently? How can process information be gathered and used for process improvements through extended services (e.g. cogeneration of citation graphs, machine-learning, etc.)?

Reviewing as well as editorial work in the *SQM* subprocess includes checking and at least partially reading the author's cited literature. This leads to a repetition of work as authors, reviewers and editors have to search, find and retrieve the same document for a proper citation check or further discussion about the content of the publication. The subprocess *SQM* also implies, strictly speaking, the documentation of scientific research (e.g. lab books, videos, recordings, printouts of measurement instruments, software, data sets) and preservation of all work that the publication refers to and is based upon. Because of different subscription contracts not all scientists involved in the publication process have access to the same literature bundle needed in the *SQM* process. Therefore, the option to submit a bundle of written document and used literature in combination with a (time-limited) access permission for all affected roles is attractive for the subprocess *SQM*.

The third subprocess *Distribution* deals with enhancements in the generation and provision of the several types of metadata. To possess an increasing data basis of submissions and corresponding literature that already might be classified can support applications and research in the fields of automatic classification, indexing, clustering and linking of related literature.

In the last subprocess *Consumption* questions arise how literature management tools can be integrated in the cycle of retrieving literature and submitting it to the LitObject middleware. Especially current approaches to import literature objects in literature management tools such as Zotero or Mendeley are mostly based on individual web-crawlers for each publication website system that have to be updated after every minor update in the website structure of the publisher. The idea of individual web-crawlers has already been pursued in the research project UniCats (presented by Lockemann et al. 2000) where a "wrapper generator" supported the development.

A more appealing approach would be to embed information directly into the website such as DublinCore tags, ContextObjects in Spans (CoIns) or Highwire Press Tags to name a few. However, in the current development state they either have no support to link a reference to fulltext documents, are lacking proper transformations to common reference data formats such as RIS or BibTeX, have ambiguous fields or do not support the description of various documents on one page such as the description of a website for a collection of papers. They are good to promote information about the element in question on the homepage but, at the moment, they should not serve as a solid metadata basis for usage in scientific publications. The main problems in this process are the error-prone and often irreversible transformations between the different metadata formats as stated by Hull et al. (2008, p. 7).

3 The LitObject Middleware

In response to the challenges of Sect. 2.1 we propose the LitObject middleware as a system component which automates the transport and transformation of literature objects as shown in Fig. 2. The key idea is that the LitObject middleware acts as a central system with webservices between the various literature management tools and digital libraries as well as a repository for the publishing process (Fig. 1).

The system as presented in Fig. 2 has three main system boundaries: The first subsystem boundary is the local literature management software (illustrated by two literature management software applications). The second subsystem is the middleware, the third subsystem is an extended service. In this paper we describe a system for the cogeneration of graphs in Sect. 4 as an example for an extended service.

Our solution to the problems of Sect. 2.1 is to keep the organization of literature on an individual level with local literature management software. The local literature management is extended by a plugin that adds an export of literature objects to the middleware (as presented in Fig. 3). The exported literature objects can be acted on (e.g. display or edit) by a website that accesses the LitObject middleware. Through the website literature objects can be imported and exchanged by local literature management systems, digital libraries and extended services.

To avoid the creation of duplicates with every import from the website and to provide not only an import, but also a notification mechanism that automatically detects changes on the side of literature management tools as well as the server side, the plugins could be extended by a synchronization mechanism as indicated by the dotted line in Fig. 3.

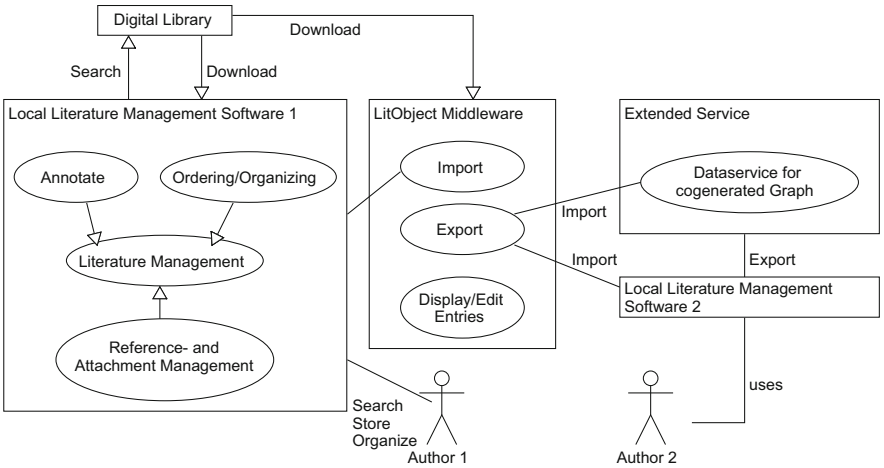

Fig. 2 System diagram of a middleware for literature objects

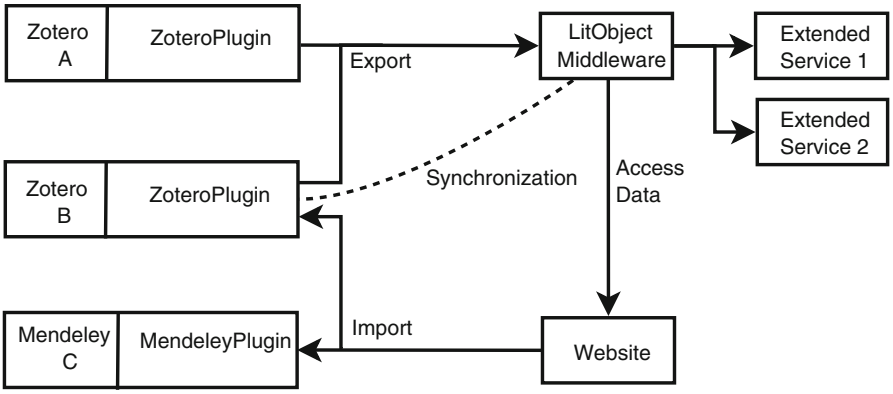

Fig. 3 Component structure for the LitObject middleware

The LitObject middleware follows the design pattern of a resource oriented architectural style as described by Fielding (2000). The main concepts of a resource oriented architecture (RESTful architecture) are a client server model with a separation between a client that requests data from a server utilizing the HTTP protocol (Fielding and Reschke 2014b), statelessness that means not to handle any session context on the server side, resource identification, a uniform interface, self describing messages and hypermedia. According to Pautasso and Wilde (2011) a resource is everything that is "relevant for an application (and its state)". Resources have an identifier resolved by a uniform resource identifier (URI) (Fielding and Reschke 2014a). Resources have a representation for their data format. Usually the Java Script object notation (JSON) or the extended markup language (XML) are used for this purpose. A client interacts with a resource through its representation by using the methods of the HTTP protocol: GET for retrieving, PUT for updating, POST for creating and DELETE for deleting resource elements. The application state is handled by using links within the representation that guide the usage of the webservice.

The LitObject middleware follows the introduced constraints. We have identified two main resources (see Fig. 4 for a detailed overview), the `items` resource that hosts literature objects and the `collections` resource which serves as a named collection for a set of corresponding literature objects. Both the `items` and the `collections` resource have instances (with identifiers) that are depicted by `<item-id>` and `<collection-id>` in Fig. 4. The `item` resource has three subresources `refersto`, `bibtex` and `attachments`. The `refersto` resource is itself a link list to other literature objects. The literature object is described by its reference at the resource `bibtex`. Fulltext documents that are part of the literature object are described and hosted at the `attachments` subresource. The `collections` resource, on the other hand, is a list of links to one or more literature objects identified by a URL. A plugin for a local literature management

http://<host>:<port>/<api_user>	- base URL of the LitObject middleware
/items	- list of all literature objects
/<item-id>	- URL of one single literature object
/refersto	- list of links to other literature objects
/bibtex	- reference of the literature object in the BibTeX format
/attachments	- list of all documents of the literature object
/attachments/<id>	- URL of one single (fulltext) document of the literatur object
/attachments/<id>/content	- content/download of the fulltext document
/collections	- list of all collections
/<collection-id>	- URL of one single collection which is a list of links to URLs of <item-id> resources

Fig. 4 URL structure of the LitObject middleware

application, like Zotero or an extended service, interacts with the URL structure via a JSON representation utilizing the HTTP verbs GET, POST, PUT and DELETE.

4 Cogenerated Citation Networks with the Help of the LitObject Middleware

In Sect. 3 we have introduced the LitObject middleware that serves as a generic service to interconnect various literature management applications and extended services. One example of an extended service for the LitObject middleware is the provision of a technical infrastructure for cogenerated citation networks.

There exist various citation link networks such as the Arxiv HEP-PH citation graph (Stanford University 2003) and also service interfaces to request link databases, for example, the ArnetMiner system as presented by Tang et al. (2008) The approach for building up the data set in the ArnetMiner system is to extract researcher profiles at first, then querying databases with the researchers' name as identifiers and storing these information in a database. An alternative approach is followed by the LitObject middleware by utilizing the exported literature objects of the (locally) organized literature. Three use cases can be distinguished that are supported by a naming convention for collections following the scheme `<collection/paper-name>:<linktype>`.

1. The author's own paper: What did he cite: `<collection-name>:cited`
2. The author's retrieval process: What literature did he find relevant for the topic? This is a latent construct: `<collection-name>:relevant`
3. The author follows the citation structure in papers. How does he record the citations he followed: `<paper-read>:citationfollowed`

As depicted in Fig. 5, at first authors organize their literature with the help of a literature management software such as Zotero following the naming convention

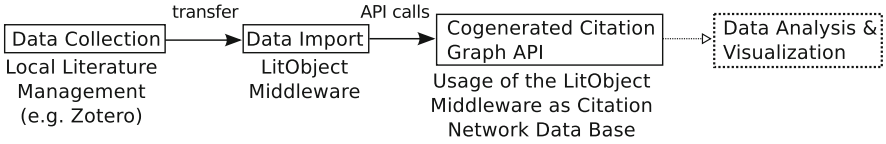

Fig. 5 Steps for utilizing the LitObject middleware as data basis for citation networks

introduced above. In the second step, they transfer their literature objects to the LitObject middleware. As shown in Fig. 4 there are two main resources in the LitObject middleware: one for the literature objects (`items`) and the second one for collections (`collections`). The extension for the literature management software exports at first all literature objects to the `items` resource and in a second step creates the same link structure as the local managed collections at the `collections` resource on the LitObject middleware side. As these imported literature objects as well as their arrangement in collections are saved in a database in a structured manner their structure is exposed by an API and—following the REST paradigm of various representations of resources—in different data formats, for example, the pajek data format or as comma separated values (csv) data.

5 Discussion

Other extended services—for example, as depicted in Fig. 3 on top of the component "Website"—are at least technically possible, e.g. the generation of an organization-wide search index or an internal repository. Also the LitObject middleware can serve as foundation for scientific projects in the fields of information retrieval, machine learning or clustering.

Key reasons to continue the development on the LitObject middleware are that the usage of the commercial sharing and collaboration features of Mendeley, Zotero, etc., limits research capabilities, forces vendor lock-in and exposes research activities permanently as well as reduces interoperability between various literature management tools.

Additionally there are legal as well as technical challenges: Within the sphere of "fair use" the exchange of full literature objects is allowed. However the restrictions of fair use are ambiguous, differ between countries and are non-uniform between publishers. Already the exchange of (digital) literature objects within one organization unit is sketchy as some Libraries advice against the sharing of literature as for instance the Health Science Library of the University of North Carolina (2014). In this technically focused publication a deeper exploration is beyond the scope, however, an exhaustive evaluation of these topics by country specific lawyers would be desirable.

On the technical side as stated by Hull et al. (2008), there still exist issues how to identify literature objects globally as not every object has a digital object identifier

(DOI), international standard book number (ISBN) or a uniform resource name (URN). The questions how to get metadata and in which representation is solved technically, for example, by the unAPI specification (Chudnov et al. 2006b) that is described and motivated more in detail in an article by Chudnov et al. (2006a). Unfortunately, unAPI is lacking the linkage to fulltext documents, however, only a small adaptation is necessary to support this requirement as well. Finally, citation network analysis and visualization services (e.g. Pajek) can utilize this interface.

6 Summary

In this paper we have introduced the LitObject middleware that is a RESTful web service to interconnect various literature management software tools. The aim of the LitObject middleware is to improve collaborative literature work for advanced publishing processes. The LitObject middleware offers the possibility to import and export literature objects. Literature objects are the combination of a reference in a data format such as BibTeX and one or more fulltext document that belong to the reference. Additionally, literature objects can also be grouped together in collections. On top of the LitObject middleware various extended services are possible. As one possible extended service we presented an application for the cogeneration of citation networks. With the help of many authors who import literature objects into the LitObject middleware it is possible to build up a network of citations and clusters of literature that belong to a particular topic.

References

Chudnov, D., Back, G., Binkley, P., Celeste, E., Clarke, K., D'arcus, B., et al. (2006b). unAPI version 1. Accessed September 23, 2014, http://unapi.info/specs/unapi-version-1.html

Chudnov, D., Binkley, P., Summers, E., Frumkin, J., Giarlo, M. J., Rylander, M., et al. (2006a). Introducing UnAPI. *Ariadne*, (48). http://www.ariadne.ac.uk/issue48/chudnov-et-al/

Fielding, R. T. (2000). *Architectural Styles and the Design of Network-Based Software Architectures*. Ph.D. thesis, University of California, Irvine.

Fielding, R. T., & Reschke, J. (2014a). Hypertext transfer protocol (HTTP/1.1): Message syntax and routing. *RFC 7230 (Proposed Standard)*.

Fielding, R. T., & Reschke, J. (2014b). Hypertext transfer protocol (HTTP/1.1): Semantics and content. *RFC 7231 (Proposed Standard)*.

Gilmour, R., & Cobus-Kuo, L. (2011). Reference management software: A comparative analysis of four products. *Issues in Science and Technology Librarienship, 66*. doi:10.5062/F4Z60KZF.

Glänzel, W., & Schubert, A. (2004). Analyzing scientific networks through co-authorship. In H. Moed, W. Glänzel, & U. Schmoch (Eds.), *Handbook of quantitative science and technology research* (pp. 257–276). Dordrecht: Kluwer Academic Publishers. doi:10.1007/1-4020-2755-9_12.

Hensley, M. K. (2011). Citation management software: Features and futures. *Reference & User Services Quarterly, 50*(3), 204–208.

Howison, J., & Goodrum, A. (2004). *Why Can't I Manage Academic Papers Like MP3s? The Evolution and Intent of Metadata Standards*. Working Paper, Institute for Software Research, Carnegie Mellon University. http://www.repository.cmu.edu/isr/494/

Hull, D., Pettifer, S. R., & Kell, D. B. (2008). Defrosting the digital library: Bibliographic tools for the next generation web. *PLoS Computational Biology, 4*(10). doi:10.1371/journal.pcbi.1000204.

Lockemann, P., Christoffel, M., Pulkowski, S., & Schmitt, B. (2000). UniCats: ein System zum Beherrschen der Dienstevielfalt im Bereich der wissenschaftlichen Literaturrecherche. *IT - Information Technology, 42*(6), 34–40.

Mead, T. L., & Berryman, D. R. (2010). Reference and PDF-manager software: Complexities, support and workflow. *Medical Reference Services Quarterly, 29*(4), 388–393. doi:10.1080/02763869.2010.518928.

Pautasso, C., & Wilde, E. (2011). Introduction. In E. Wilde & C. Pautasso (Eds.), *REST: From research to practice* (pp. 1–18). New York: Springer.

Solomon, J. (2009). Programmers, professors, and parasites: Credit and co-authorship in computer science. *Science and Engineering Ethics, 15*, 467–489. doi:10.1007/s11948-009-9119-4.

Stanford University. (2003). High-energy physics citation network. Accessed September 23, 2014, https://www.snap.stanford.edu/data/index.html#citnets

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., & Su, Z. (2008). ArnetMiner: Extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Las Vegas)* (pp. 990–998). New York: ACM.

University of Chicago Press. (1982). *The Chicago manual of style* (13th ed.). Chicago: The University of Chicago Press.

University of North Carolina Health Sciences Library. (2014). Copyright basics: Sharing articles - LibGuides at University of North Carolina Chapel Hill. Accessed September 23, 2014, http://www.guides.lib.unc.edu/c.php?g=9031&p=45264