Towards New Scholarly Communication: A Case Study of the 4A Framework

Pavel Smrz and Jaroslav Dytrych

Brno University of Technology
Faculty of Information Technology
Bozetechova 2, 612 66 Brno, Czech Republic
E-mail: {smrz,idytrych}@fit.vutbr.cz

Abstract. This paper discusses the use of semantic web technology to realize the vision of future scientific publishing and scholarly communication. It introduces a novel knowledge system that builds on the popularity of social tagging and collaboration systems as well as on the idea of "Annotations Anywhere, Annotations Anytime" (hence 4A). Key technical characteristics of the realized components are presented. User experience observations and the results of a preliminary experiment are also reported.

1 Introduction

Recent years brought various proposals for changes in the model of scientific publishing and scholar communication. Most radical ones call for a complete shift from pre-publication peer review and impact factor measures towards social popularity of papers [13, 1]. Moderate views, e. g., [15, 16], consider how the success of community-specific services (arXiv[1], ACP[2]) could be replicated in other fields.

The 4A framework introduced in this paper adds a piece to the mosaic of tools that enable such changes. It can be seen as a modularization and unit-testing system in the parallel between scientific knowledge artifacts and software artifacts [6]. Pieces of knowledge on a very low granularity level (e. g., particular ideas) being annotated play the role of modules or units in this parallel. Testing corresponds to validation of annotations in collaborative knowledge building and to community appreciation of automatically generated summaries, inference-based re-uses, and other applications of the semantically enhanced resources.

Transformations in scholar communication are often motivated by benefits they could bring to a research community. On the other hand, business models applicable in the new publishing era need serious consideration [21]. Publishers obviously try to find their position in the evolving environment. It can be demonstrated by their effort to engage authors and get their feedback or to support sharing of scientific citations among researchers in the form of reference manager

[1] http://arxiv.org/
[2] http://www.atmospheric-chemistry-and-physics.net/

services (e. g., Connotea[3] or CiteULike[4]). The 4A framework is also intended to encourage the publishers to see opportunities in making money by adding value to open access data, rather than restricting the access to outcomes of research [9]. Publishers could build on their experience in organizing bodies of experts (e. g., editorial boards and review teams) able to identify the quality of the content. By means of advanced knowledge processing tools, they could extend these competences to trace expertise in specific domains, to aid the knowledge structuring process, its sharing and reuse, to support the community work around research topics, and to measure the impact of individual as well as group activities.

The framework emphasizes the role of annotation in the process of knowledge creation. It is crucial to realize that tags can be associated not only with a scientific publication, but also with any other material relevant for the research. If the experimental data referred to in a paper is made available (as required in some disciplines today) and also annotated, it is easier to verify its interpretation presented in the text. The annotation of data that is employed in experiments by other researchers also simplifies tracing its reuse and thus provides an alternative to the current state characterized by reusing and not appraising [7].

A special attention has been also paid to the means of tracking down particular ideas described in papers to the source code of computer programs described in the text. One can employ the concept of program documentation generation (for example, by means of the popular Doxygen tool[5]) and interconnect particular pieces of code with the description of their functionality on the higher level.

Last but not least, the concept of annotation is general enough to cover not only the textual material related to the presentation of a particular research result (slides from talks, blogs, tweets or other formats of messages referring to the results) but also related audio- or video recordings. For example, the proof-of-concept case study involved a material from the Speaker and Language Recognition Workshop Oddysey 2010[6] and it employed the search-in-speech service Superlectures[7]. Many ideas from referring texts are not easy to link to the material available in textual form so that working with a multimedia content is often necessary.

To summarize the directions of motivation mentioned in this section, the new scientific publishing and scholarly communication model should be accompanied by an annotation framework supporting the whole life-cycle of scientific papers – from ideas, hypotheses, identification of related research, and data collection, through setting and running experiments, implementing solutions and interpreting results, to submitting, reviewing, reflecting reviews, preparing final versions, publishing complementary material, getting feedback, discussing content and reflecting previous results in a new work. Annotations on all levels pave

[3] `http://www.connotea.org/`
[4] `http://www.citeulike.org/`
[5] `http://www.stack.nl/~dimitri/doxygen/`
[6] `http://www.speakerodyssey.com/`
[7] `http://www.superlectures.com/odyssey/`

the way for shared knowledge understanding. The social dimension of ubiquitous annotations of knowledge artifacts can also bring immediate benefits to research communities in terms of better models of fine-grained impact characteristics. The elaborated annotation system helps to pinpoint an expertise in particular fields. Researchers can also benefit from instant gratification – the annotation is immediately available for others, it can be shared and re-used in other contexts.

This paper discusses the use of semantic technologies to realize the above-mentioned vision. The name of the annotation framework – 4A – refers to the concept of "annotation anywhere, annotation anytime". Key features of the realized solution are introduced in Section 2. A proof-of-concept implementation of the 4A annotation client and its application in experiments are presented in Section 3. The paper concludes by discussing the directions of future work.

2 The 4A framework

Semantic search and other advanced functionality of the future web offer clear benefits for the semantic annotation of knowledge artifacts. On the other hand, the semantic enrichment of resources presents a tedious work for users. It is therefore crucial to lower the barrier to annotate by means of immersing the annotation into everyday work of users.

We address this requirement by defining a general interaction schema and protocols that can be supported in various contexts. An ultimate goal of the 4A framework is therefore to let users annotate naturally in any application used. The current implementation focuses mainly on the textual resources and implements the functionality of server components as well as several clients.

From the annotation support perspective, it is crucial to distinguish two types of environments users work in viewer- and editor applications. As viewers do not modify the source content, changes in annotations need to be transferred only. On the other hand, it is tricky to synchronize editing annotation sessions as the changes in text may invalidate annotations.

To serve both types of clients, a general annotation exchange protocol has been defined. The 4A synchronization server implements the protocol and enables coordinated work of 4A clients. An annotation extension to general Javascript-based editors has been developed as the first 4A client. A PDF reader add-on and a Firefox browser extension are being implemented.

The 4A framework goes beyond the current practice of simple keyword tagging and knowledge structuring curated in advance. It introduces an intuitive knowledge structuring schema of structured tags [5]. In the experiments, it has been successfully used for describing necessary conditions, conflicting views, and comparison patterns. The 4A clients present structured tags in the form of a relation attribute tree.

Unfortunately, none of the annotation formats applied in existing tools is general enough to suit our purposes without modifications. Even Annotea [10] – a format resulting from a W3C initiative to standardize annotations – cannot express relations among structured annotations. That is why we could not simply reuse an existing format. The 4A format extends Annotea by introducing structured annotations with attributes of various types, embedded annotations and interrelations among annotations. It is based on RDF where the subject is always the annotation. It includes: annotation ID (URI), type, time of creation, its author, URI of an annotated document (or its server copy), XPath to an annotated textual fragment, its offset, length and textual content, annotation content and a specification of annotation attributes. The RDF Schema corresponding to the RDF model is available at `http://nlp.fit.vutbr.cz/annotations/rdfs/annotation-ns.rdf`.

The position of an annotated fragment is given by the path in the document object model (DOM), the offset and the size. The representation is robust to changes in general formatting. It is usually not necessary to process the whole document. For example, a web page boilerplate and other parts that are not in a DOM node on the path to an annotated fragment will be ignored.

The annotation types form a hierarchical structure. They include common comments (a note, a description) and basic types of entities (a thing, a person, etc.). Users can add new types and create complex type hierarchies.

The URI of an annotated document identifies a copy of the document that is stored on the server. The annotation process starts with a synchronization step in which the client sends the document URI and its content to the server. The server returns the URI of the local copy of the document which will be used in annotations. This procedure enables annotating documents that the server could not access directly. Processing the original document on the server side also enables removing irrelevant attributes (e. g., session ID) and applying changes to the correct version of a document as the stored version is updated together with all annotations at every access.

A new annotation interchange protocol has been defined for the communication between 4A clients and the server(s). In addition to actual annotations, it can be used for simple authentication, synchronization of annotated documents, subscription to annotations from defined sources, annotation suggestions, interchange of knowledge structures (annotation and attribute types) and various annotation-related settings.

The protocol enables two-way asynchronous communication between clients and servers. If a user adds an annotation, the server sends it immediately to all other users that annotate the same document and are subscribed to a given channel (defined by an author, a group or an annotation type). Changes of annotation types, of the document content and of relevant settings are distributed immediately as well.

Messages are defined in XML. They can be therefore easily sent over various protocols on a lower level and parsed on the client side. It is also possible to combine the messages into one XML and make the communication even more efficient.

Session management messages include the protocol version negotiation, logins and log-outs. Subscription management enables specifying what types of annotations from what sources should be sent to a particular client. Annotations can come from another user or a URI representing an automatic annotation server, user group or other general source (e. g., an external service).

The server gets a copy of the current version of an annotated document by means of a document synchronization process. If it is the first time the document is sent, the server just stores it. If there is already a copy of the document, the server compares the new version with the stored one and updates the stored version together with all annotations. If the new version impacts no previous annotation, the operation is confirmed instantly. If there are annotations that could be invalid, the server informs users who can consequently correct possible errors.

To support clients that are not able to work with structured texts, the server can linearize the text from documents. The client transforms the document to plain text and sends it to the server. If there is a structured form of the document at the server, it is linearized and compared to the received version. If they are the same, it is possible to start the annotation process. The server will then adapt all incoming annotations for the structured version of the document. The linearization also enables cooperating with clients working with other structured formats than the server. The client will adapt positions to the linearized text, the server will adapt it to its structured form. This way it is possible to annotate the same text synchronously e. g. in HTML and in PDF.

3 Proof-of-Concept Experiments

As a proof of concept, we employed the current version of the JavaScript annotation editor in experiments . The tool is a universal component which can be easily integrated into various JavaScript-based editors such as TinyMCE[8]. It implements the functionality of a 4A client – it enables editing complex annotations, synchronizes tagging with the server side and presents annotation suggestions provided by information extraction components.

The client makes the annotation process manageable. The type of annotation can be specified in a text field. Instant search in type names is supported – the input serves as an intelligent filter. It is also possible to browse the hierarchical structure of types. This reduces diversity of annotation types.

It is possible to add new attributes to identified relations. A name, a type and a value are associated with each attribute. The selection of an attribute type is similar to that of the annotation type. The way information is presented also corresponds to the types. An attribute can be of a simple data type, of an extended type (e. g., a geographic location), or of an annotation type. Simple and extended data types are added as new branches of the type tree. If an annotation type is utilized, it is possible to choose one of the existing annotations (an annotation reference) or to create directly a nested annotation.

It is possible to select more textual fragments in a document. Annotating more fragments simultaneously enables identifying all occurrences of a relation

[8] http://tinymce.moxiecode.com/

in a document. When storing it, a separate annotation is generated for each fragment. The same procedure can be applied for attributes (nested annotations) with the same name, type and content which are then displayed as a list. Clients employ suggestions to facilitate the tagging process (see Figure 1 for an example).

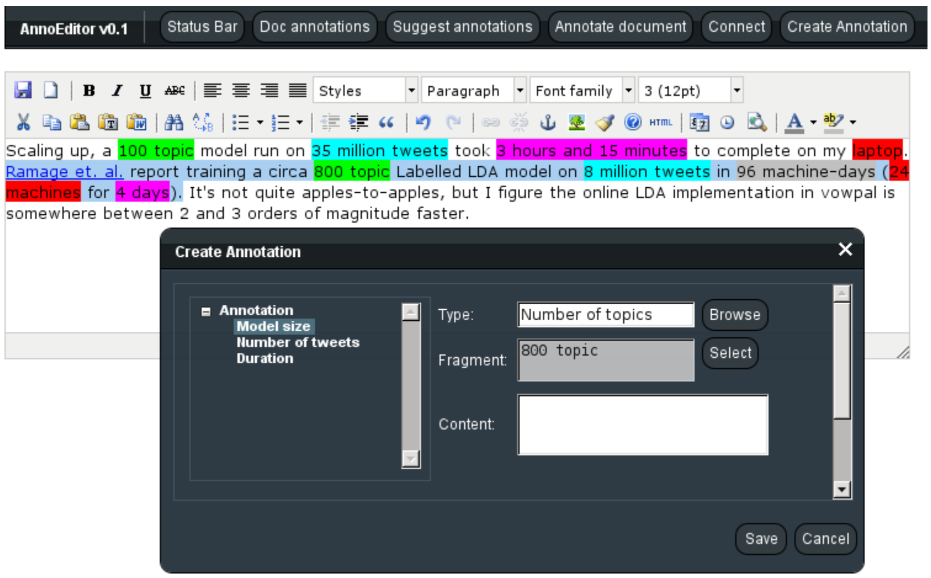

Fig. 1. Suggesting relation attributes based on the previous context

One senior researcher (the first author of this paper) and six PhD students (including the second author) participated in the annotation experiments. Source texts included not only scientific papers, but also blog messages and tweets related to specific research topics and the material referred from these sources (e. g., a particular part of a dataset, a PowerPoint presentation, a source-code file). The papers dealt mainly with natural language processing, information extraction and machine learning (correspondingly to the expertise of the participants). However, there were also general topics discussed (especially in the blog messages).

The experiments aimed at identifying particular pieces of text (selecting textual fragments) that correspond to specific types of content patterns (e. g., statements showing equivalence of two approaches or suggesting a resource for a specific group of readers). A part of the task also involved tagging the fragments supplying evidence that a particular method is really applied for a particular task (not just referred to as an alternative the paper does not really deal with). For citations, the task was to find the part of a referred text illustrating the referring context.

The primary objective of the analysis was to learn what form of tags (simple or structured ones) users prefer in given situations and how tag suggestions help to annotate the content (even if they are far from being perfect). Annotation suggestions based on automatic information extraction were switched off to not influence the answer to the first part of the question. This setting simulates specialized tagging where the set of available annotated examples is too limited to be used for learning-based methods or too complex to be described by a handful set of rules. The participants have been instructed that the annotation should be usable for future knowledge acquisition from annotated resources. Suggestions were turned on for the second part. Figure 2 shows a typical setting of the environment.

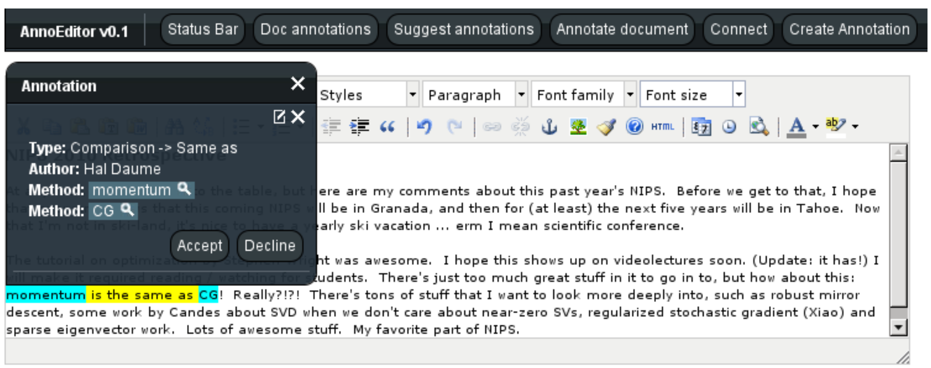

Fig. 2. Structure tag suggestions

Results of the first part (annotation form preference) are conclusive. All annotators resort to structured tags in the task of method comparison. On the other hand, they prefer flat tags over structured ones when dealing with simple annotations of individual methods ("conjugate gradient" is a "ML method"). As a side effect of the advanced functionality of the tool, it was also observed that the annotation form will probably stay unchanged if a user defines the structure of a tag for a specific task and others are able just to follow and re-use the definition.

The second part of the question turned to be hard to answer. Automatic annotation suggestions accelerate the tagging process but they also distract annotator's attention. The level of acceptance is subjective and significantly varies with respect to the precision and recall of the information extraction process generating suggestions. It is also questionable what the granularity of structured-tag suggestions should be, i. e., what information (that does not need to be correct) should be combined and potentially confirmed by one click. For example, one of the annotators pointed out that "momentum" should not be referred to as "a method" in the context shown in Figure 2 so that he could not accept the suggestion as a whole.

4 Related Work

A lot of previous work has been done in fields related to the presented research. The general topic of future research, scientific publishing and scholarly communication is discussed in various contexts – linking scholarly literature with research data [3], advocating open access [8], demonstrating a role of social media in scholarly communication [4], etc.

First tools supporting general annotations on the web date back to mid-nineties [11, 18]. Among current solutions, Annozilla[9] is conceptually the closest to 4A browser add-ons – it is realized as an extension to the Firefox browser, tags are stored on local or remote servers, XPointer is used to identify the annotated part of a document. The annotation protocol developed in the Annotea project [10] is employed. In contrast to the 4A framework, Annozilla is intended for simple tagging only. A pre-defined set of annotation types is limited to general categories such as a comment, a question, an agreement/a disagreement. It is therefore not possible to use the tool for advanced knowledge structuring.

PREP Editor [14] and Bundle Editor [24] can illustrate innovative ideas in the area of interweaved text authoring and tagging. The former represents one of the first real-time collaborative text writing tools. Implemented annotations are limited to this functionality. The latter enables annotation structuring by means of grouping them into bundles. Annotations can be filtered, sorted etc. Zheng [23] states that the structured annotations proved to be more efficient and user-friendly than the simple ones.

Popular web-based editors and other office applications such as Google Docs[10] or Microsoft Office Live[11] show current trends in the development of collaborative tools. Even though there is either no or very limited annotation functionality, they need to be considered as "opinion makers" in terms of user interface simplicity.

Google Docs is also noteworthy as being a successor to Google Wave[12] – an envisioned distributed platform of Email 2.0 where clients should be connected to their particular servers and the servers should communicate by means of the Google Wave Protocol[13]. Compatible solutions, such as Novell Vibe[14], only started to appear when Google turned away from a strong support of the platform. Nevertheless, the vision of a distributed platform transferring user actions with a very low granularity stimulated the development of the 4A framework.

---

9 http://annozilla.mozdev.org/
10 https://docs.google.com
11 http://www.officelive.com/
12 https://wave.google.com
13 http://www.waveprotocol.org/
14 https://vibe.novell.com/

The area of social tagging is also relevant for general topics discussed in this paper. Various tools exist for the specific subdomain of collaborative scientific citation managers such as CiteULike, Connotea, Bibsonomy[15], or Mendeley[16]. The systems offer advanced support for the particular task (automatic extraction of relevant metadata, export to various formats, etc.) but they are usually not able to link references to particular pieces of text and do not explicitly deal with knowledge structuring.

On the other side of the knowledge processing support scale, there is a family of ontology, topic maps and other knowledge representation format editors. Protégé[17] defines a kind of standard, other tools such as Neon Toolkit[18], Ontolingua[19], or TopBraid Composer[20] stress the collaborative-, user support-, or integration aspects, respectively. Anchoring created knowledge structures in real data is often limited to examples or glosses that can be stored together with concepts and relations. In this context, resources resulting from semantic annotation efforts in computation linguistics, such as FrameNet[21] or OntoNotes[22], as well as enriching folksonomies by formal knowledge sources [2] are relevant for our research.

The presented work also extends the concept of semantic wikis. Several systems have been created around the idea of semantic web technologies enhancing the wiki way of content creation [12]. The 4A framework directly draws on our experience from the development of the KiWi system [17]. For example, the knowledge emergence approach based on structured tags gains from [5] resulting from the KiWi project.

Last but not least, one of the key components of the 4A framework – the information extraction module – relates to previous research on automatic knowledge extraction. In particular, we take advantage of the KiWi extraction elements [20, 19] that employ general purpose solutions such as Gate[23] or GeoNames[24]. In spite of the fact that information extraction does not form a main topic of this paper, relevant solutions for semi-automatic learning of ontologies from text such as TextToOnto[25] and OntoGen[26] or ontology-based information extraction [22] need to be mentioned as a source of inspiration.

[15] `http://www.bibsonomy.org/`
[16] `http://www.mendeley.com/`
[17] `http://protege.stanford.edu/`
[18] `http://neon-toolkit.org`
[19] `http://www.ksl.stanford.edu/software/ontolingua/`
[20] `http://www.topquadrant.com/products/TB_Composer.html`
[21] `http://framenet.icsi.berkeley.edu/`
[22] `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2008T04`
[23] `http://gate.ac.uk/`
[24] `http://www.geonames.org/`
[25] `http://texttoonto.sourceforge.net/`
[26] `http://ontogen.ijs.si/`

5 Conclusions and Future Directions

The 4A framework presented in this paper incorporates annotation into knowledge acquisition and knowledge sharing processes. The proof-of-concept implementation and the use of an annotation client in tagging experiments proved validity of the idea but also revealed imperfections of the model and flaws in the user interface of the realized tools. The future work will focus on removing these deficiencies. The concept of knowledge emergence and knowledge structuring needs to be refined. We will flesh out the support for social knowledge-creation processes. New information extraction modules will employ advanced machine learning methods to provide better suggestions in situations when only a few training examples are available.

The number of 4A clients will also grow. The development of the JavaScript annotation component and its integration into various web-based editors will continue. The Firefox browser extension and the PDF reader add-on will be finished. We will initiate the work on other extensions for LibreOffice and the Semantic Desktop. The annotation format will be proposed as an extension of the official W3C Annotea system.

New experiments will explore advanced tagging collaboration patterns. The question on the acceptable error-rate for the automatic suggestions will be also tackled. A higher number of participants and a more advanced setting of experiments are necessary to prove the efficiency of the knowledge structuring processes in the 4A framework. Last but not least, consistency and adequacy of the knowledge representation resulting from the use of the 4A tools will be studied.

Acknowledgements

The research leading to these results has received funding from the European Community's 7th Framework Programme FP7/2007-2013 under grant agreement number 270001 – Decipher.

References

1. Adler, B., de Alfaro, L., and Pye, I. Redesigning scientific reputation. *The Scientist 24*, 9 (Sept. 2010), 30. Online: http://www.the-scientist.com/article/display/57645/.
2. Angeletou, S., Sabou, M., and Motta, E. Improving folksonomies using formal knowledge: A case study on search. In *ASWC* (2009), pp. 276–290.
3. Attwood, T. K., Kell, D. B., McDermott, P., Marsh, J., Pettifer, S. R., and Thorne, D. Calling international rescue: knowledge lost in literature and data landslide! *Biochemical Journal 424*, 3 (2009), 317–333.
4. Bilder, G. Social media and scholarly communication. In *ISMTE European Conference, International Society of Managing and Technical Editors* (2010).

5. Bry, F., and Kotowski, J. A social vision of knowledge representation and reasoning. In *SOFSEM 2010: Theory and Practice of Computer Science*, J. van Leeuwen, A. Muscholl, D. Peleg, J. Pokorny, and B. Rumpe, Eds., vol. 5901 of *Lecture Notes in Computer Science*. Springer, 2010, pp. 235–246.
6. Casati, F., Giunchiglia, F., and Marchese, M. Liquid publications: Scientific publications meet the web, 2007. Online: https://dev.liquidpub.org/svn/liquidpub/papers/deliverables/LiquidPub%20paper-latest.pdf.
7. Enriquez, V., Judson, S. W., Weber, N. M., Allard, S., Cook, R. B., Piwowar, H. A., Sandusky, R. J., Vision, T. J., and Wilson, B. Data citation in the wild, 2010. DOI: 10.1038/npre.2010.5452.1, Online: http://precedings.nature.com/documents/5452/version/1.
8. Furnival, A. C. Open access to scholarly communications: advantages, policy and advocacy, 2011. Online: http://eprints.nottingham.ac.uk/1419/.
9. Hall, M. Efficiency and effectiveness: Digital futures in innovation, Oct. 2010. Presentation at the JISC Future of Research Conference on 19th October 2010.
10. Kahan, J., and Koivunen, M.-R. Annotea: An open RDF infrastructure for shared web annotations. In *Proceedings of the 10th International Conference on World Wide Web* (2001), ACM, pp. 623–632.
11. LaLiberte, D., and Braverman, A. A protocol for scalable group and public annotations. *Comput. Netw. ISDN Syst. 27* (Apr. 1995), 911–918.
12. Lange, C., Schaffert, S., Skaf-Molli, H., and Völkel, M., Eds. *4th Semantic Wiki Workshop (SemWiki 2009) at the 6th European Semantic Web Conference (ESWC 2009), Hersonissos, Greece, June 1st, 2009. Proceedings* (2009), vol. 464 of *CEUR Workshop Proceedings*, CEUR-WS.org.
13. Moody, G. Abundance obsoletes peer review, so drop it. http://opendotdotdot.blogspot.com/2010/06/abundance-obsoletes-peer-review-so-drop.html.
14. Neuwirth, C. M., Kaufer, D. S., Chandhok, R., and Morris, J. H. Computer support for distributed collaborative writing: defining parameters of interaction. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work* (New York, NY, USA, 1994), CSCW '94, ACM, pp. 145–152.
15. Nielsen, M. The future of science, 2008. Online: http://michaelnielsen.org/blog/the-future-of-science-2/.
16. Poeschl, U. Interactive open access publishing and public peer review: The effectiveness of transparency and self-regulation in scientific quality assurance. *IFLA Journal 36*, I (2010), 40–46. DOI: 10.1177/0340035209359573, Online: http://www.atmospheric-chemistry-and-physics.net/pr_acp_poschl_ifla_journal_2010_interactive_open_access_publishing.pdf.
17. Schaffert, S., Eder, J., Grünwald, S., Kurz, T., Radulescu, M., Sint, R., and Stroka, S. KiWi – A platform for semantic social software. In *Proceedings of the 4th Workshop on Semantic Wikis, European Semantic Web Conference* (2009).
18. Schickler, M. A., Mazer, M. S., and Brooks, C. Pan-browser support for annotations and other meta-information on the world wide web. In *Proceedings of the fifth international World Wide Web conference on Computer networks and ISDN systems* (Amsterdam, The Netherlands, The Netherlands, 1996), Elsevier Science Publishers B. V., pp. 1063–1074.
19. Schmidt, M., and Smrz, P. Annotation component for a semantic wiki. In *Proceedings of the 5th Workshop on Semantic Wikis, European Semantic Web Conference* (2010).
20. Smrz, P., and Schmidt, M. Information extraction in semantic wikis. In *Proceedings of the 4th Workshop on Semantic Wikis, European Semantic Web Conference* (2009).

21. Waltham, M. Why does one size not fit all in journal publishing?, 2010. Presentation at the June 2010 Society for Scholarly Publishing meeting, slides available online: `http://www.marywaltham.com/SSP_Seminar_2010.pdf`.
22. Wimalasuriya, D. C., and Dou, D. Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science 36*, 3 (2010), 306–323.
23. Zheng, Q. Structured annotations to support collaborative writing workflow. Master's thesis, The Faculty of Graduate Studies (Computer Science), The University of British Columbia, Dec. 2005. (accessed online March 14, 2011).
24. Zheng, Q., Booth, K., and McGrenere, J. Co-authoring with structured annotations. Department of Computer Science, University of British Columbia, 2006.