

Multi-Layer Active Documents for the Semantic Web

Vitaveska Lanfranchi, Fabio Ciravegna

Department of Computer Science,
University of Sheffield
Regent Court, 211 Portobello Street
S1 4DP Sheffield, UK
E-mail: {vita|fabio}@dcs.shef.ac.uk

Daniela Petrelli

Department of Information Studies
University of Sheffield
Regent Court, 211 Portobello Street
S1 4DP Sheffield, UK
E-mail: d.petrelli@shef.ac.uk

INTRODUCTION

Users' disorientation and cognitive overload are well known phenomena intrinsic to the idea of hypertext and studied since the early days [3]. The Semantic Web (SWeb in the following) with its layers of annotations can increase the cognitive overload when a document is accessed. As a matter of fact, in a SWeb framework, annotations can be added at different stages in a document lifetime. Initially annotations are added at editing time by the author [11] [12] [13]. Annotations can span from tagging portions of documents with concept labels, to identifying instances or concept mentions, to connect information (e.g. a telephone number and its owner). Different users can annotate the same document in different ways, e.g. using different ontologies, creating different views of the same document. Other annotations can be composed at reading time, i.e. when the document is displayed, for example added by automatic semantic harvesters that extract and integrate information from different repositories [14] [15], or systems which link entities to additional information or services [4].

The many times a document can be annotated and the many pieces of knowledge potentially connected can easily transform a SWeb document into an intricate set of connections. Moreover, the semantic consistence of the annotations (e.g. outgoing links) cannot be guaranteed when different heterogeneous schemas can be applied to the same document. For example, from the author of a document it would be possible to add a hyperlink to reach her diary; from her contact address it is possible to reach the weather forecast for that region. Though both annotations are perfectly acceptable in principle, it is likely that such different navigation choices would distract and disorient the user.

This paper proposes to organize the annotations into layers to offer functionalities specific to the user and the context of use as a way to limit the cognitive overload. Managing layers of annotations requires the document to be active. An active document is aware of its own content and can flexibly change the way it presents itself to the actual user, e.g. by allowing the user to read only up to a predefined

level of detail. However an active document should not be limited to the presentation phase: we extend the activity to the annotation layer as well. A global framework for editing and accessing multi-layered semantically active documents is proposed.

DEFINING A MULTI-LAYER DOCUMENT

The idea of layers was first introduced in the late 80s and has been recently revamped by Ben Shneiderman [7] who proposes multi-layer design to reduce interaction complexity. He focuses on multi-layer interfaces, different interfaces with an increasing number of features of increasing complexity to accommodate users learning stages. McGrenere, Baecker and Booth [6] applied the idea of layers to personalization: the user decides which features should be available at every layer.

We apply the idea of layers to annotated documents: a document can contain different layers of information that can be edited, visualized and used at different times or by different users. In other words different use(r)s could require the presentation or generation of different layers. For example, easier reading could be supported with most of annotations hidden; A selective display of a subset of annotations would allow navigating the information about a specific topic; The full information would offer full access to the SWeb potentialities. Also, a novice user could use only some basic functionality (present at a lower layer), while an expert user could exploit the full structure (higher layers). We propose to support these different modalities of access by organizing the annotation into multiple layers in order to facilitate both the writing and reading experience.

SEMANTICALLY ACTIVE DOCUMENT

Active documents have autonomous behaviour [1] and can be active in many ways: they can be able to act independently and autonomously, reacting to environment changes, they can have their own goals and plans and follow some strategies [11]. An active document is able to adapt itself to the context and to the user [12]: when an active document is displayed on a PDA the style is different than when is displayed on a desktop screen. Active documents also cooperate between them and with the user [5].

We propose *semantically active documents*, an extension of the concept to the SWeb. In our vision, a semantically active document is able to exploit SWeb technologies to support users during all the document lifecycle. Semantically active documents know their annotations and are able to deal with the information contained. They are able to learn from the interaction with both other documents and users; by monitoring user's writing they can support authorship showing new relationship and proposing new annotations, they can highlight mistakes and offer corrections. Finally, an active document knows its content and is able to automatically update the information used to compose it changes.

Semantically active documents must provide a way to structure the overabundant information available when using SWeb technologies, for example by filtering the content accordingly to the appropriate layer or the user preferences. We propose a classification of the information using abstract classes. Example of classes are:

- Business information
- Geographical information
- Semantic information (dictionary service and translation services)
- Personal information

The information should be grouped into layers and presented considering user profiles and the context in which it is used or visualized. For example, in supporting a doctor writing on her clinic website to announce that the vaccination against influenza is available, she would welcome a definition and some additional information to be linked. On the other hand when information about influenza vaccination is added to a patient record, the system should not propose any generic definition.

A SCENARIO

To better explain the framework described above, a possible scenario of use is outlined in this session.

An institution has to provide an annual report. Despite being used only once a year, a predefined template of its structure already exists. Each section has an expected author associated, e.g. the responsible will write the overview, the accountant will fill in the tables of the final balance and the projection for the coming year, the secretary will add changes in the personnel section. The semantically active document can check that the person filling a certain part has the rights to do it and can also solicit those contributions from the authors as the deadline approaches.

The accountant is filling in the final balance session: the system has located last year annual report and proposes the user to connect this year summary table with the projection compiled the year before. Similarly the table containing the budget for the coming year is connected with data located outside the document and that can be used by the active document to automatically update its content.

In the front the list of the personnel is active and links names to personal pages. Should a person leave, the active document will try to keep the document consistent by tracing the person in the new position.

Different parts of the document contain different degree of sensitive information. This is captured by the layers that determine how the document should be presented to who, or in other words, which parts of the document should be opened to which reader. A public section is unrestricted and its content can be freely distributed, e.g. to journalists. Other parts of the document, e.g. projections and plan for the coming year, are instead encrypted and accessible only to those who know the key. Depending on the reader's profile, the semantically active document is able to identify the level of details that that reader can access thus automatically selecting the default display layer.

A FRAMEWORK

We propose a framework for managing semantically active documents that support users during the creation, editing and visualisation of multi-layer documents. The system helps users accomplish their task, acting as a personal information management system. Information automatically gathered from other documents is made available to the user in a multilayered way. Filters to select the appropriate information describe the possible classes of information, the possible generic user profiles and the possible context of interaction. Intelligent agents are in charge of monitoring the user, the information classes, and the context. In this way, users select each time the desired class of information and the system calculates the best pieces of information to show. The preferred information sources are recorded in the user profile and used at this time by the information harvester for retrieving. An ontology describes the process of document manipulation during these three phases and the needs of services for each phase. Another (partition of the) ontology describes the application domain relevant to the document and the annotation schema. The system we envisage provides modalities of use for both the creation phase and the display phase of a document.

The Creation Phase

We envisage the following main phases in the lifecycle of a semantically active document:

- drafting;
- writing;
- publishing
- maintaining.

Below we describe these phases and the system's contribution in creating the document and its annotation.

Drafting

When drafting, the general content of the document is outlined, detailing the different parts of the document and

their likely content¹. Expected contribution by the system is the proposal of an initial template structure for the new document and its division into layers. Each layer has its own set of functionalities and the system is able to tune its interface to the required features. In addition a template to control the visualization has to be provided at drafting time. Users can modify the templates (structure, layers and visualization), create their own or move from a layer to another one.

Writing

When the initial outline is completed, the system should proactively find relevant or related content to suggest. Interesting information, hyperlinks and relevant citations will be provided. More importantly, the system should provide semantic annotation by using the many methods available in the SW field, e.g. automatic annotators like in SCREAM [12] and MnM [13] or harvesters [14].

Publishing

Once a document is edited it is ready to be divulged. The system should provide the author a way of controlling the diffusion of the document. The author should decide if the document can be modified or not, who could do it and when. As an example, the author sets that a document can be modified by few colleagues but only till a deadline. A decision has to be taken on which layers of the document are available and for whom. In fact an author can decide that the basic layer is publicly available while higher layers are only for a restricted number of users (e.g., colleagues). This feature is complemented by public key cryptography systems to protect private layers of the document.

A document can in principle link other documents that have reading restrictions (e.g., for subscribers). There is an issue on how to control document diffusion. A multi-layer active document can automatically verify if the use done is coherent with the specified privacy preferences. These preferences regulate the way the document is displayed and affect the interface. If the document is read-only, no editing options or commands will be offered but the interaction will be limited to navigate the annotation, surf the SWeb, follow hyperlinks, use new services and so on.

Maintaining

The maintenance phase involves the aftermath of the document finalization and delivery. Currently the documents lifecycle ends with the delivery. The only way to update is to modify them manually. Using SWeb technologies it is possible to provide support for semi- or automatic update. There are a number of relevant pieces of

information that are useful after finalization. For example for scientific papers relevant information includes submission, acceptance, publication details, referees' comments, citations received, related relevant papers on the same topic (e.g. new papers that cite some of the bibliography cited in the current paper as well). Most of this information could be collected in an automatic way. For example Citeseer is able to provide most of them; a harvester able to wrap Citeseer and add semantic to it would be also able to add this information.

More complex situations could be envisaged that would allow exploiting the power of the Semantic Web. Special functionalities could be associated to parts of documents to allow automatic update. For example some tables and graphs could be produced using information from the Web and could be updated automatically by the system every time the document is opened.

The Display Phase

When the document is accessed, filters are applied on the available annotation (and related services) to select the appropriate information to be displayed on the basis of the classes of information, the user profiles and the context of interaction. Semantic classes of information can be complemented with other information, e.g. about authorship. For example a user could choose to visualize only the annotations written by an expert colleague as opposed to a novice, or only those inserted by an expert human. Preferences could not only derive from an explicit user profile but also emerge from tracing user's behaviour.

We propose to perform visualization using XML technologies where the layers are visualized using XSLT templates. Different CSS files can be applied to change the visualization with respect to user preferences.

OPEN QUESTIONS

The motivation of this proposal is to simplify the view and use of the SWeb. However there is an intrinsic complexity in a multi-layer interaction as the user has to deal with different views of the document, of the interface, and of the semantic network. Probably an expert user would exploit this possibility of having different semantic views, but a novice could have difficulties in learning how to use the tool and in understanding the multi-layer architecture. This will be a matter of investigation through usability testing.

The reliability in retrieving and intrusiveness in proposing new information should also be considered. The acceptable level of intrusiveness is an open question worth researching.

ACKNOWLEDGMENTS

This work was carried out within the Dot.Kom project (www.dot-kom.org), sponsored by the EU IST as part of Framework V (grant IST-2001-34038), and the AKT project (www.aktors.org), sponsored by the UK Engineering and Physical Sciences Research Council (grant GR/N15764/01).

¹ We consider here the case of a single self-contained document (e.g. a newspaper article). In a Web environment the concept of document is fuzzier, as information is generally spread across a group of pages. In this case, the outline action refers to the design of the organization of information across the group of pages rather than within the single page.

REFERENCES

1. Bompani, L., Ciancarini, P., Vitali, F. Active documents in XML. In *ACM SigWeb Newsletter* 8, 1 (1999) 27-32.
2. Ciravegna, F., Dingli, A., Petrelli, D., Wilks, Y. User-System Cooperation in Document Annotation based on Information Extraction. In Asuncion Gomez-Perez, V. Benjamins, R. (eds.) *Knowledge Engineering and Knowledge Management (Ontologies and the Semantic Web), Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management* (1-4 October 2002, Sigüenza, Spain), Lecture Notes in Artificial Intelligence 2473, Springer Verlag, 2002.
3. Conklin, J. Hypertext: an introduction and survey. In *IEEE Computer*, 20, 9 (September 1987), 17-41.
4. Domingue, J. B., Dzbor, M., Motta, E. Semantic layering with Magpie. In Staab, S., Studer, R (eds.) *Handbook on Ontologies in Information Systems*, Springer Verlag, 2003
5. Heinrich, E., Maurer, H. Active Documents: Concept, Implementation and Applications. In *Journal of Universal Computer Science* 6, 12 (2000), 1197-1202.
6. McGrenere, J., Baecker, R., Booth, K. An evaluation of a multiple interface design solution for bloated software. (2002) In *Proceedings of Human Factors in Computing System – CHI02*, ACM Press, New York, 164-170.
7. Shneiderman, B. Promoting universal usability with multi-layer interface design. In *Proceedings of the ACM conference on Universal usability* (Vancouver, Canada), ACM Press, New York, 2003, 1-8.
8. Wang Baldonado, M.Q., Woodruff, A., Kuchinsky, A. Guidelines for using multiple views in information visualization. In *Proceedings of the working conference on Advanced visual interfaces* (Palermo, Italy), ACM Press, New York, 2000, 110-119.
9. Werle P. Hansson K. and Kilander F. Agentifying information: Active documents to Support Work Processes in Heterogeneous and Distributed Environments. In *Proceedings of the PCC Workshop* (November 29-30, Lund), 1999.
10. Werle, P., Jansson, C.G. Active Documents Supporting Teamwork in a Ubiquitous Computing Environment. In *Proceedings of the PCC Workshop* (April 3-5, Nynäshamn), 2001.
11. Goble C., Bechhofer S., Carr L., De Roure D. and W. Hall: Conceptual Open Hypermedia = The Semantic Web? In *Proceedings of the Second International Workshop on the Semantic Web*, Hong Kong, May 2001
12. S. Handschuh, S. Staab, and F. Ciravegna. S-CREAM-Semi-automatic CREATION of Metadata. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*. Springer Verlag, 2002.
13. Vargas-Vera M., Motta E., Domingue J., Lanzoni M., Stutt A., Ciravegna F.: MnM: Ontology driven semi-automatic or automatic support for semantic markup. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*. Springer Verlag, 2002.
14. Ciravegna F., Chapman S., Dingli A., and Wilks Y.: Learning to Harvest Information for the Semantic Web, in *Proceedings of the European Semantic Web Symposium*, Heraklion, Greece, May 2004
15. Glaser H., Alani H., Carr L., Chapman S., Ciravegna F., Dingli A., Gibbins N., Harris S, schraefel m.c., Shadbolt N.: CS AKTiveSpace: Building a Semantic Web Application in *Proceedings of the European Semantic Web Symposium*, Heraklion, Greece, May 2004