FROM WYSIWYG TO WYSIWYM – CONTENT AND VALUE ENRICHMENT WITH SEMANTIC METADATA

Daniel Hladky, Victor Klintsov, Ali Khalili, Sören Auer

National Research University – Higher School of Economics, Moscow, Russia
dhladky@hse.ru, vklintsov@hse.ru

Institute für Informatik, AKSW Universität Leipzig, Leipzig, Germany
khalili@informatik.uni-leipzig.de, auer@informatik.uni-leipzig.de

Abstract. Many companies strive to increase their value proposition to the traditional Web search engine and to novel applications. With the increased popularity of the semantic web and Linked Open Data this paper is presenting a method to create rich semantic annotations using the RDFaCE approach. The approach is based on providing different views to the content authors such as a classical WYSIWYG view and a WYSIWYM (What You See Is What You Mean) view making the semantic annotations visible.

1. Introduction

In 1985 Porter proposed the value chain as a tool to identify and to analyze the origins of competitive advantage (Porter n.d.). With the advent of the Web and especially with the Semantic Web (Berners-Lee et al. 2001) the power of buyers has changed due to the availability of public available information. Companies enhance activities in order to increase the value proposition (Figure 1).
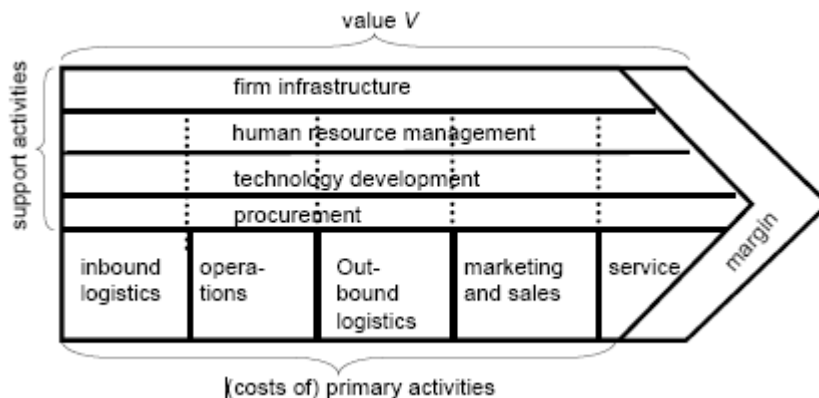

Figure 1 – Value Chain

The enhancement is mainly to create machine readable information that can influence Web search engines and novel applications by enriching Web content with semantic annotations. Use cases are Search Engine Optimization (SEO), technologies to improve better links to content[1] or to enable e-commerce[2]. Based on a survey performed by Avicomp, Ontos[3] and Peter Mika[4] (see figure 2) show that 75-80% of the validated web pages insufficient results against HTML/CSS and semantic metadata but confirms the trend of RDFa.
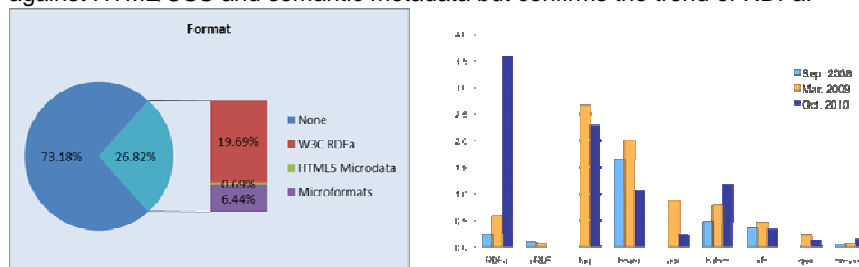

Figure 2 – Metadata Validation and RDFa trend

In this paper we present the RDFaCE approach for combining WYSIWYG text authoring with the creation of rich semantic annotations which is part of Content Management Systems. The contributions of this work are in particular:

---

[1] http://open.blogs.nytimes.com/2012/02/16/rnews-is-here-and-this-is-what-it-means/

[2] http://semtechbizuk2011.semanticweb.com/sessionPop.cfm?confid=63&proposalid=4385

[3] http://avicomp.ru and http://ontos.com

[4] http://tripletalk.wordpress.com/2011/01/25/rdfa-deployment-across-the-web/

- An overview of Semantic Content Authoring (SCA) process.
- The architecture of an RDFa authoring environment called RDFaCE (RDFa Content Editor).
- Describing News domain and "rNews" as a use case for RDFaCE.

2. Semantic Content Authoring

In the context of this paper, we define *semantic content authoring (SCA)* as the tool-supported manual composition process aiming at the creation of documents which are based on a non-semantic representation form (e.g. text or hypertext) enriched with semantic representations during the authoring process. Semantic documents facilitate a number of important aspects of information management:

- For *search and retrieval*: help to create more efficient and effective search interfaces, such as faceted search (Tunkelang 2009) or question answering (Lopez et al. 2011).
- In *information presentation:* help to create more sophisticated ways of flexibly visualizing information, such as by means of semantic overlays as described in (Burel et al. 2009).
- For *information integration:* provide unified views on heterogeneous data stored in different applications by creating composite applications such as semantic mashups (Ankolekar et al. 2007).
- To realize *personalization:* provide customized and context-specific information which better fits user needs and will result in delivering customized applications such as personalized semantic portals (Sah et al. 2007).
- For *reusability* and *interoperability:* facilitate exchanging content between disparate systems and enables building applications such as executable papers (Müller et al. 2011).

3. RDFaCE System Architecture

The RDFaCE system architecture is depicted in Figure 3 and consists of three layers. The foundation layer on which we ground the RDFaCE plugin includes the *TinyMCE Rich Text Editor*[5]. This open source HTML editor was chosen because it is very flexible to extend and is used in many popular Content Management Systems (CMS). The RDFaCE includes the following components:
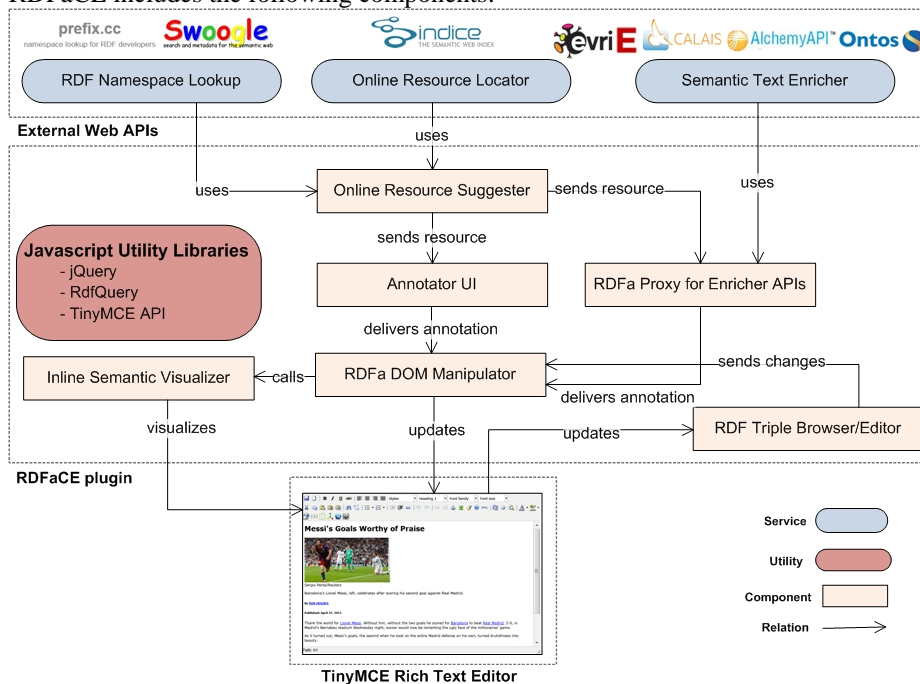

Figure 3 – RDFaCE System Architecture

Annotator UI: This component uses the TinyMCE API as well as jQuery UI to provide user friendly interfaces for RDFa content editing. As shown in Figure 4, the normal annotation procedure consists of four steps: 1) Defining appropriate namespaces. 2) Selecting a fragment of the text. 3) Assigning the subject (and type) to be used for the selected fragment. 4) Inserting triples by assigning properties. After annotations are received by users, they are delivered to RDFa DOM manipulator.

RDFa DOM Manipulator: This component is responsible for manipulating the Document Object Model (DOM) according to the desired RDFa annotation. The component adds the required metadata into the HTML code.

Inline Semantic Visualizer: The main goal of the inline semantic visualizer is to provide a kind of on-demand visualization which can be included/excluded on the fly within the WYSIWYG content editing. Each time a new annotation is added by RDFa DOM manipulator, this component is called to visualize the editor.

---

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

RDF Triple Browser an Editor: This component extracts the RDF triples embedded in the text and provides the edit and delete functionality for these triples. When user edits or deletes a triple, these changes are delivered to RDFa DOM manipulator to update the content correspondingly.

Online Resource Suggester: This component provides the user with a set of accessible online resources. In order to perform this task, it accesses a number of external Web APIs. The Online Resource Suggester works in a close relation to Annotator UI. It facilitates the task of annotating content by searching the terms which are selected by user and suggesting corresponding URIs.

RDFa Proxy for Enricher APIs: This component acts as a proxy to make the output of enricher APIs (i.e. NLP text annotation services) consumable as RDFa. Extracted entities are mapped against a desired vocabulary in order to make appropriate annotations. These annotations are delivered to the RDFa DOM manipulator to update the content correspondingly.
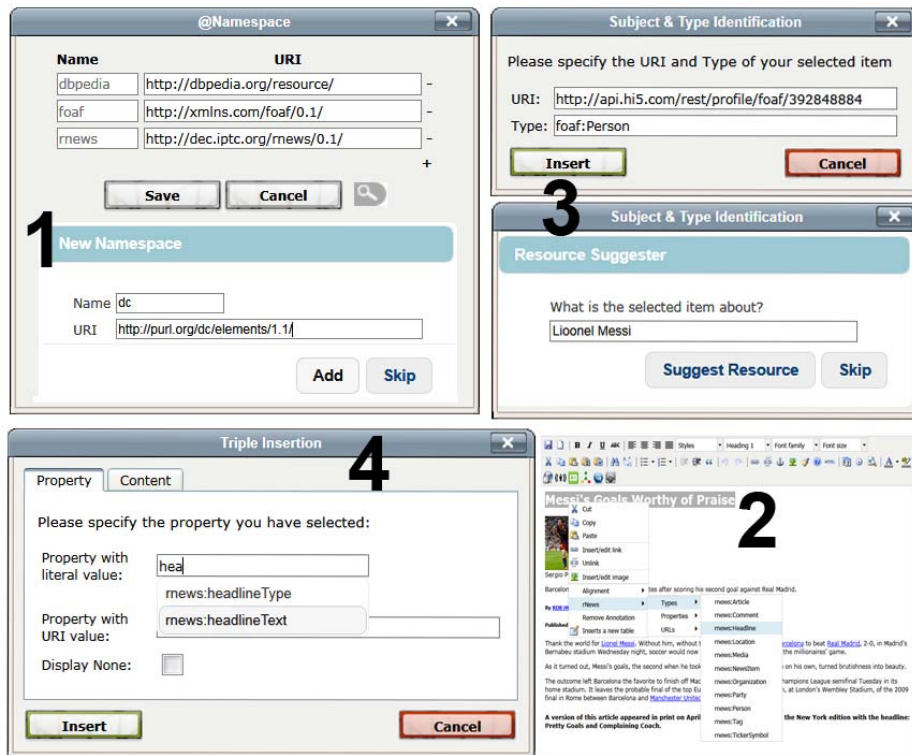
Figure 4 – Annotation user interface

4. Views for Semantic Text Authoring

The main innovation of RDFaCE is the support of different views on the semantically annotated content. RDFaCE supports four different views for semantic text authoring, which are shown in Figure 5 and explained briefly in the sequel. The user can easily switch between these views and even use them in parallel. The views are syncronized so that applying changes in one of the views automatically updates other views.
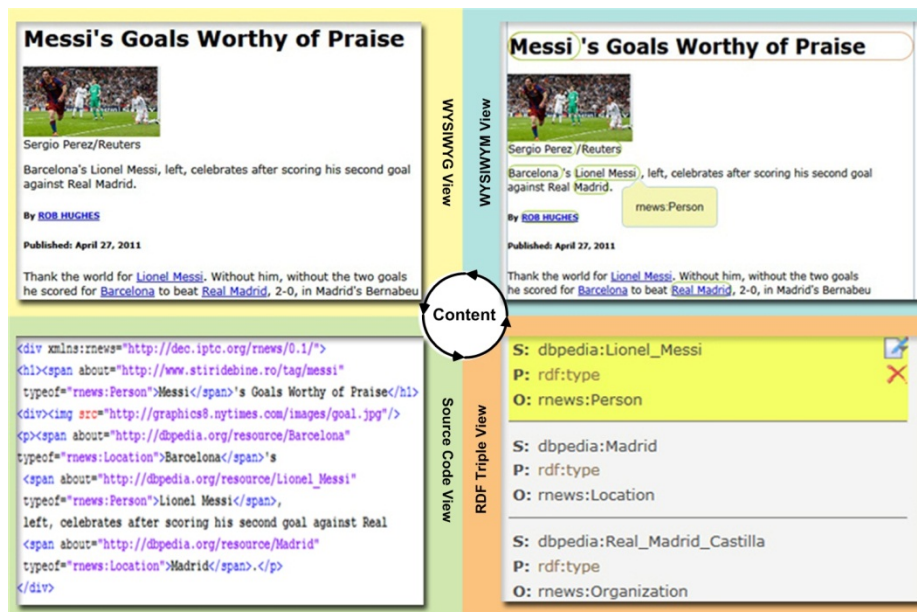
Figure 5 – Views for semantic text authoring

WYSIWYG View: The What-You-See-Is-What-You-Get view is the classical interface for rich-text authoring within main CMS systems and used by authors.

WYSIWYM View: The What-You-See-Is-What-You-Mean view is an extension of the WYSIWYG view, which highlights named entities and other semantic information. When pointing with the mouse on a highlighted annotation RDFaCE shows additional information concerning the particular annotation as a tooltip.

RDF Triple View: This view summarizes all the facts, which can be extracted from the annotated text. It provides a deeper semantic view when compared to WYSIWYM view. Since the triple view reveals all the triples embedded in the text, it can be called as WYMIWYS (What-You-Mean-Is-What-You-See) view.

Source Code View: Finally, the source code view shows the HTML source of the article including the RDFa annotations. This view is primarily intended for software engineers supervising the publication workflow as well as knowledge engineers.

5. Use Case "News"

Following example shows a use case from the news industry using TinyMCE in connection with RDFaCE and the usage of rNews[6]. rNews is a proposed standard by IPTC[7] for using RDFa to annotate HTML documents with news specific metadata. The general production process of news involves several steps as shown in Figure 6. The initial step is linked to crawling of external and internal sources (Hladky et al. 2007) helping to gain an inside into stories. The next step consists of writing and editing of the article based on the usage of the Web-CMS which is using TinyMCE and RDFaCE. Once an article is written and approved by the chief-editor the article is being processed using the described solution in section 2 and 3. Before the final publishing to the Web the content including the semantic annotations are being written to the archive and the RDF store for further usage such as:

- Recommendation Service: A plugin that shows relevant articles using filters based on concepts like person, organization or location.
- Breaking-News Ranking: Applying an algorithm that uses metadata and Web statistics in order to create automatic suggestions for the front page (breaking news).

---

[6] http://dev.iptc.org/rNews

[7] http://www.iptc.org/site/Home/
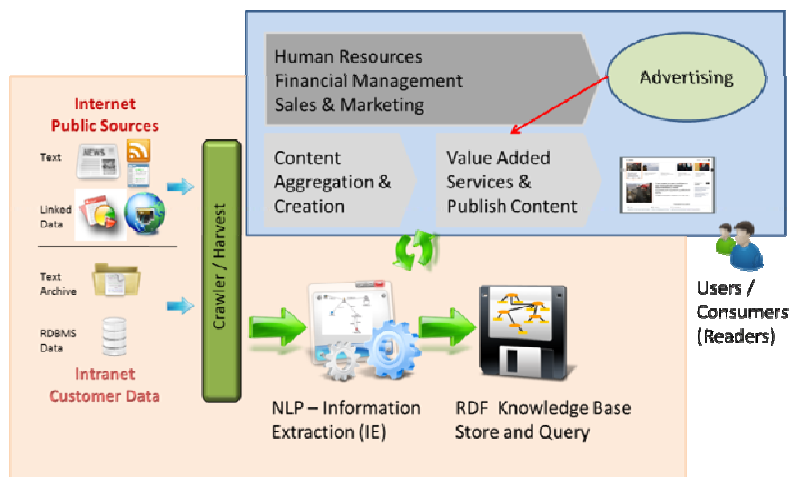
Figure 6 – News Value Chain

References

Ankolekar, A. et al., 2007. The two cultures: Mashing up Web 2.0 and the Semantic Web. *Web Semantics Science Services and Agents on the World Wide Web*, 6(1), pp.70-75.

Araujo, S., Houben, G.-jan & Schwabe, D., 2010. Linkator : Enriching Web Pages by Automatically Adding Dereferenceable Semantic Annotations. *Source*, pp.355-369.

Auer, S., Dietzold, S. & Riechert, T., 2006. OntoWiki – A Tool for Social , Semantic Collaboration I. F. Cruz et al., eds. *Science*, 4273(4273), pp.736-749.

Berners-Lee, T., Hendler, J. & Lassila, O., 2001. The Semantic Web K. Aberer & K.-S. Choi, eds. *Scientific American*, 284(5), pp.34-43.

Burel, G., Cano, A.E. & Lanfranchi, V., 2009. Ozone Browser: Augmenting the Web with Semantic Overlays. In C. Bizer, S. Auer, & G. Aa. Grimnes, eds. *5th Workshop on Scripting and Development for the Semantic Web*. pp.3–4.

Efimenko, I., Hladky, D. & Klintsov, V., 2008. Semantic Technologies and Information Integration: Semantic Wine in Media Wine - skin Presentation Roadmap. *SciencesNew York*.

Hladky, D., Ehrlich, C. & Khoroshevsky, V., 2007. Social and Semantic Web Technologies: Semantic Navigation, Digesting and Summarization. *Knowledge Management*.

Di Iorio, A. et al., 2010. Ontology-driven generation of wiki content and interfaces. *New Review Of Hypermedia And Multimedia*, 16(1), pp.9-31.

Lopez, V. et al., 2011. Is question answering fit for the Semantic Web? A survey. *Semantic Web*, 2(2), pp.125-155.

Müller, W. et al., 2011. A-R-E: The Author-Review-Execute Environment. *Procedia Computer Science*, 4, pp.627-636.

Porter, M.E., *Competitive Strategy - Techniques for Analyzing Industries and Competitors*, The Free Press.

Sah, M. et al., 2007. SEMPort – A Personalized Semantic Portal. *Human Factors*, pp.4-5.

Siorpaes, K. & Simperl, E., 2009. Human Intelligence in the Process of Semantic Content Creation. *World Wide Web Internet And Web Information Systems*, 13(1-2), pp.33-59.

Spivack, N., 2010. The semantic web: from representation to realization. *Transactions on computational*, pp.90–107.

Tunkelang, D., 2009. *Faceted Search* G. Marchionini, ed., Morgan & Claypool Publishers.

Uren, V. et al., 2006. Semantic annotation for knowledge management: requirements and a survey of the state of the art. *Web Semantics Science Services and Agents on the World Wide Web*, 4(1), pp.14-28.