

Semantic Web Interfaces for Newspaper Multimedia Content Management

Ferran Perdrix^{1,2}, Roberto García², Rosa Gil², Marta Oliva² and José A. Macías³

¹Diari Segre. Del Riu 6, 25007 Lleida, Spain. fperdrix@diarisegre.com

²Universitat de Lleida. Jaume II 69, 25001 Lleida, Spain. {rgarcia,rgil,oliva}@diei.udl.cat

³Universidad Autónoma de Madrid. Tomás y Valiente 11, 28049 Madrid, Spain. j.macias@uam.es

Abstract The S5T project explores the possibilities of a semantic content management system in the context of a mass media group (radio, TV, newspaper and internet). The system is based on Semantic Web technologies and attempts to build up a media repository that makes the media house more productive. To carry out such a challenge, the project features a user interface that takes profit from all the modelling and annotation effort, which is carried out during ontology construction and semantic annotation of audio transcriptions. In particular, three interfaces have been developed whose goal is to facilitate the main interaction modes, search and browsing. For the former, a query-by-example tool has been created. For the latter, there are two specialised interfaces, which allow browsing media contents and the semantic annotations for transcriptions. These interfaces make it possible to take profit from the underlying ontologies during user interaction.

Introduction

Recently, many newspaper companies are changing into news media houses. They own radio stations and video production parties that create content unsupported by traditional newspapers, but that they are able to be delivered by Internet newspapers. Web-based news publishing is evolving fast, as many other Internet services, and nowadays this service is trying to adapt information in manners that better fit user's interests and capabilities in the digital space.

The S5T¹ project is looking forward to producing more profit from news sites using cutting-edge ways to archive multimedia content based on Semantic Web

1. Scaleable semantic personalised search of spoken and written contents on the Semantic Web (S5T), <http://nets.ii.uam.es/~s5t>

technologies and human speech recognition. Previously, the Neptuno project (Castells et al, 2006) handled semantically enriched textual news and explored new capabilities in terms of information retrieval and content browsing. Now, in S5T, the objective is to expand this benefits to audio and audiovisual content in order to integrate the many kinds of multimedia content that are managed in a media house, such as radio or TV news.

In order to reuse existing semantic annotation tools and expand their benefits to audiovisual content, the approach followed in the S5T project is to transcribe the human speech present in the audio of radio and TV news. Once speech is recognized and transcribed as text, it is possible to annotate it reusing many of the tools in place for textual content.

Users exploit the new capabilities that the semantic web technologies provide while they search and browse the news repository in a transparent way. In order to accomplish this in a usable and accessible way, specialized user interfaces are required. Consequently, the aims of the S5T project range from aspects concerning content creation, indexation based on human speech and search to multimedia content visualization, assisted query authoring and browsing.

A use case

SEGRE² is a mass multimedia company that owns one TV channel, two radio stations, one newspaper and a web portal. SEGRE produces every day 64 newspapers pages, 8 hours of self-produced TV content, 24 hours of direct Radio emission and updates the web portal every day. Nowadays, contents are archived separately by media channel: TV, Radio, Internet and newspaper. Identifying relationships among contents from different repositories is a cumbersome task, and the effort needed to manually integrate all these contents is not justifiable in terms of cost-benefit scale. Consequently, SEGRE is a good use case to test the improvements proposed in this project. SEGRE has a lot of content in several media: audio, video, text, etc. and can provide audience for the two groups of interest related to this project: journalists and end-users.

Architecture

The S5T project is based on domain conceptual models formalized using ontologies as it is shown on the left side of Fig. 1. Ontologies are used to annotate the digital contents managed by the system. In order to do this, a pre-processing task is carried out with the aim of recognizing text from the audio part of audiovisual content. During the transcription, keywords are also detected and related to the ontology concepts using lexicons like WordNet (Fellbaum, 1998).

2. Diari Segre Media Group, <http://www.diarisegre.com>

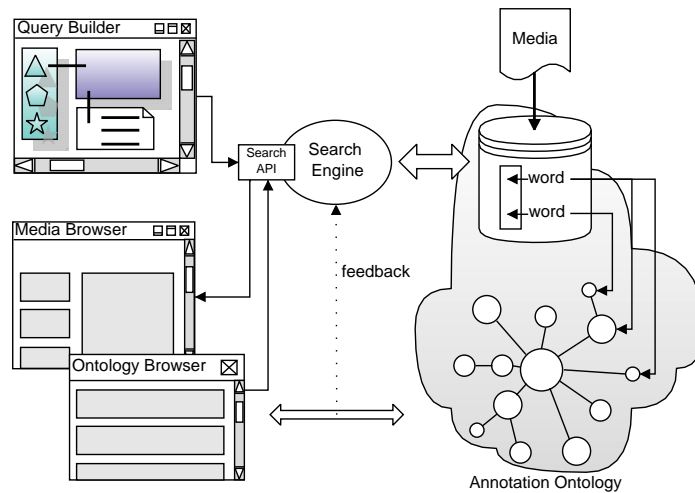


Fig. 1. System architecture

The transcription process makes it possible to integrate any kind of content with audio, and also to seamlessly integrate the annotation process with purely textual content. Then, semantic annotations make it possible to perform more accurate searches that take profit from the semantic relationship captured by the domain ontologies at place. There are more details about the transcription, ontology development and semantic annotation processes in (Tejedor et al., 2007).

However, in this paper, the focus is placed on the left side of Fig. 1. The objective is to show how the underlying ontologies can be used in the user interface in order to provide a richer experience while maintaining its complexity low and avoiding usability losses. Otherwise, the lack of a user interface that takes profit from the domain knowledge as the users interact might make many of potential benefits unreachable as long users do not pose the semantic query in the right way.

Human-Semantic Web Interaction

One of the Human-Computer Interaction's great challenges has always been to change the way computer science was originally conceived, shifting from the primitive computer-centered conception to a brand new user-centered paradigm (Shneiderman, 2003). For instance, though the Semantic Web has been accepted as one of the most promising approaches to data integration and intelligent information processing, this cutting-edge technology is still out of the scope for most of end-users. This is because users find it very difficult to learn and use applications based on Semantic Web technologies (Shadbolt, Berners-Lee & Hall, 2006).

The core of these applications is perfectly functional and the main lack might be in their scalability. However, there is a clear lack of user interface tools and pa-

radigms that facilitate, in a simple and usable way, to search and browse semantic metadata and the underlying web ontologies. Therefore, the Semantic Web can be considered usable just for skilled users, and even they find great difficulties as well (Heath et al., 2006). In the S5T research project, we have adopted a User Centered Design approach that promotes putting the user in the centre of the development process and tries to minimize their cognitive charge when interacting with ontologies and Semantic Web metadata.

First, we have focused our efforts on detecting the main user tasks when interacting with a multimedia repository and the underlying metadata. The main tasks, and also the main interaction modes, are query and browse. For the former, we propose a high-level query-intended authoring tool in order for end-users to extract and manage information easily. The objective is that users can pose queries in a really easy way using a query by example methodology and a usable user interface. They do not need to be an ontology-expert or have a priori knowledge about how the domain is structured.

On the other hand, there are two tools intended for browsing media as well as the semantic metadata and ontologies used to annotate it. The browsing approach is based on a methodology that attempts to facilitate browsing graph-data structures, like the Semantic Web, in a usable way. This methodology is detailed in the “Browsing Interface” section, where two user interfaces build on top of it are also presented. The first one is intended for browsing media and the audio transcriptions. The other one is intended for browsing the semantic annotations about media and their transcriptions, both the metadata and also the underlying ontologies.

S5T users start a typical interaction with the system by constructing a query that will allow retrieving the content they are interested in. The user receives assistance from the query authoring tool that guides him through the query construction phase. This allows end users building advanced queries and to manage the available information to construct queries in an easy way. Once the query has been build and sent, the results are shown through the first navigation user interface, the one that presents audiovisual contents and is called the *Media Browser*. This interface shows the contents that have been selected by the query, their audio transcriptions and all the available editorial metadata (title, genre, type, etc.).

Audio transcriptions are enriched with links to the domain ontology for each of the detected keywords. These keywords are detected during the transcription process (Tejedor et al., 2007) and, when there is a corresponding concept in the domain ontology, a link is generated using a semantic annotation. The *Knowledge Browser*, the other browsing interface, allows users following this links and retrieving information about the concepts in the ontologies used for annotating content transcriptions. The whole knowledge structure can be browsed using this interface. For instance, a politics domain ontology for news about politics.

Query Authoring Tool

In order to reduce the gap between users and the semantic web, we provide end-users with the ability of querying and visualizing the existing knowledge available in our ontological media repository. To this end, we propose a low-abstraction-level WYSIWYG authoring tool supporting end-user-intended inference and automatic customization of the retrieved information, see Fig. 2. In our tool, a user with no programming-in-query-language skills can freely manipulate high-level representations of knowledge obtained from previous queries or simply from scratch.

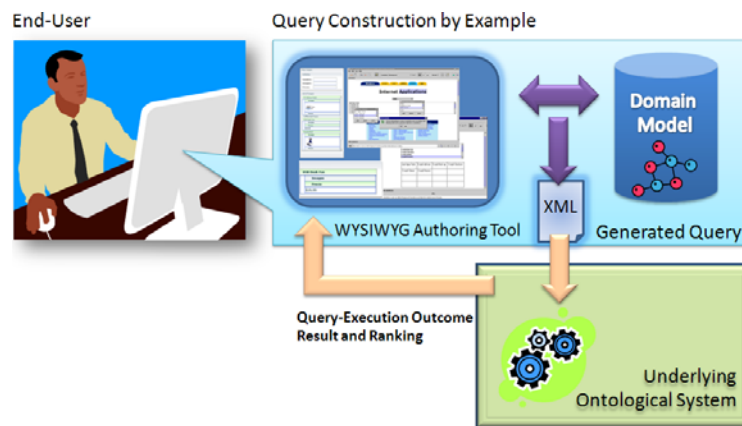


Fig. 2. Schema of the Query Authoring Tool

Continuously, the system analyses the user's interaction and the underlying interface-domain information, which is automatically updated and maintained by the system from one session to another. By using a simple visual language, the user carries out graphical queries that the system automatically transforms into a XML query-like language, in order to be processed by the underlying ontological engine. The data obtained from the query execution is processed later on to obtain appropriate feedback and update the visual information on the environment, in order for the end-user to compose far more expressive queries visually.

We based on cutting-edge End-User-Development techniques such as Programming by Example (Lieberman, 2001), which consists in automatically generalizing programmatic output from examples that the user provides to the system. From this point of view, our system can be regarded as a query-by-example tool, but featuring further improvements with respect to traditional approaches.

Queries can be applied to a great deal of data models. Originally, they appeared 30 years ago, mostly related to relational databases where the main concern was to retrieve the best information possible in the shortest time (Ozsoyoglu, 1993). Later on, query languages evolved from programmatic to other semantic notations, where the main concern was to help users compose queries easily, finally using

authoring environments such as Microsoft Query-By-Example, which is based on a previous query-by-example notation developed by IBM (Zloff, 1977). Additionally, other existing notation languages are based on the object-oriented paradigm such as ConQuer (Bloesch & Halpin, 1996) and GOQL (Keramopoulos et al., 1997), improving the expressiveness in information specification and retrieval with respect to the traditional relational model.

In our query-by-example tool, we leverage the semantic restrictions of previous models by using ontologies, which have proved to be rather flexible and expressive than other (object and relational) approaches. Our Programming by Example techniques allow for inferring user intents in real interaction with domain objects at the visual level, with the aim of both visualizing and extracting information easily (Macías & Castells, 2008).

In our environment, queries are composed by example when using the feedback provided by the underlying ontological system. This provides extra semantic information to reason about and assist users further while building queries. In a nutshell, the idea behind this is to get a real trade-off between expressiveness and easy-of-use by considering ontologies and semantic meta-information at the interface level and decreasing complexity in interacting with non-skilled.

Browsing Interface

In the Web, the basic interaction paradigm is browsing, which is based on the successive visualization of Web pages by following the links that connect them. Pages and links are the main building blocks upon which the interaction is built. Web pages are intended for human users consumption and methodologies to make the whole process usable and accessible are web established.

However, this browsing paradigm, and the principles to make the whole thing usable and accessible, cannot be directly applied to the Semantic Web. It is based on a different model, which is not based on pages and links. In the Semantic Web the fundamental block is the triple, composed by <subject, predicate, object>. This choice makes it clear that the Semantic Web was thought on different basis than the “classical” Web.

Triples are combined in order to build descriptions that try to capture the knowledge to be formalized. The combination of many triples builds up a graph; the Semantic Web data. Although the resulting model is easier to process by computers, the consumers of Semantic Web metadata are, at the end, human users. Consequently, usable and accessible interaction mechanisms are also required.

First of all, the basic browsing paradigm should change because the Semantic Web makes it very difficult to base the navigation on documents. In other word, to present to the user all the triples in a document as it is done in the Web. There is not presentation information and, what is more important, the amount of information in a single document can be too large, more than thousands of triplets. More-

over, the frontiers between documents are very fuzzy in the Semantic Web. Usually, many documents are combined in order to get coherent results as a unique graph. Thus, the problem here is where to put the limits when presenting semantic metadata to the user. In other words, how that piece is built and how new pieces are built and presented fulfilling user needs in order to build a browsing experience through the whole graph.

Semantic Web browsers like Tabulator (Berners-Lee et al., 2006) feature a more classical approach and show to the user all the triples from a Semantic Web document as an unfoldable tree. This approach embodies many usability problems because the tree grows rapidly and becomes difficult to manage. This is so because documents contain many triples and because each navigation step adds more triples from the new document to the current set.

Another approach is faceted browsing, like in /facet (Hildebrand et al., 2006). However, our preference is a simpler browsing mechanism that, though it might lack the guidance provided by facets, can deal better with heterogeneous information spaces. Moreover, it is not clear how systems like /facet deal with metadata structures that feature many anonymous resources, as it is the case for the semantic metadata managed in the S5T project.

In order to facilitate browsing, the S5T approach is based on the construction of graph fragments (García & Gil, 2006). Following this approach, it is possible to construct coherent fragment for a graph starting from any graph node that is not anonymous. For instance, for the metadata that describes a piece of content, the starting point is the node that represents it and that is the subject for all the triples that describe it. This node has an ID and consequently is not anonymous.

All the triples that start from this node are part of the fragment. To this set, all the triples that describe objects that are anonymous are also added. This happens for all nodes that are only identifiable in the context of the starting node. For instance, the for metadata describing a piece of content and the corresponding audio segment generated during the transcription process, the segments do not have an identifier and thus their descriptions will be included in the fragment describing the content item.

Consequently, all the triples that describe anonymous items are also in the fragment. This way, it is possible to construct coherent fragments that become the browsing steps. From a fragment, it is possible to follow the next browsing step starting from the triple objects that are not anonymous. The process continues iteratively and interactively, the metadata describing the selected identified node is retrieved and the fragment is built.

For instance, the user might want to get a description of the genre attribute. Therefore, he triggers a new browsing step, e.g. by clicking the genre URI, and the metadata describing it is retrieved from the IPTC News Codes Subjects Ontology³ showing that the genre label is “politics” and some additional metadata about it.

³ <http://rhizomik.net/ontologies/2006/01/NewsCodes-SubjectsOnto.owl>

Moreover, in order to make the results more usable, when they are rendered as interactive HTML the fragments are augmented with all the available labels and titles for all the appearing URIs. For instance, the left part of Fig. 3 shows the HTML rendering for some metadata fragments. As it can be observed, the URI for the genre value has been replaced by the label specified for its URI.

Media Browser

Once the query is executed, results are shown by the *Media Browser*. This interface allows navigating through the multimedia pieces selected by the query and presents the available RDF metadata describing them. These descriptions are based on a generic rendering of RDF data as interactive HTML for increased usability (García & Gil, 2006). Fig. 3 shows a view of the Media Browser interface. On the left, there are the renderings for the metadata fragments describing the retrieved contents. On the right, there is a specialized audiovisual view that renders the content, audio and video, and allows interacting with it through a click-able version of the audio transcription.


.../20070113.mp3 a AudioType	http://www.segre.com/audio/20070113.mp3
title Butlletí Nit	
date 2007-01-13	La mobilització en contra dels transgènics i en favor de Josep Pàmies també ha servit per introduir altres reclamacions. En aquest cas, alguns dels col·lectius de la lluita contra aquests cultius demanen que la Universitat de Lleida rebi una especia-
genre politics	
Referrers	

Fig. 3. Media Browser presenting content metadata (left) and the annotated transcription (right)

Two kinds of interactions are possible from the transcription. First, it is possible to click on any word in the transcription that has been indexed in order to perform a keyword-based query for all content in the database where that keyword appears.

Second, the transcription is enriched with links to the ontology used for semantic annotation. Each word in the transcription whose meaning is represented by an ontology concept is linked to a description of that concept, which is shown by the Knowledge Browser detailed in the next section. The whole interaction is performed through the user web browser using AJAX in order to improve the interactive capabilities of the interface.

For instance, the transcription includes the name of a politician that has been indexed and modeled in the ontology. Consequently, it can be clicked in order to get all the multimedia content where the name appears or, alternatively, to browse all the knowledge about that politician encoded in the corresponding domain ontology.

Knowledge Browser

This interface allows users browsing the knowledge structures used to annotate content. The same RDF data to HTML rendering used in the Media Browser is also used in the Knowledge Browser. Each browsing step through the ontologies is based on interactive metadata fragments like the one shown in Fig. 3. Consequently, continuing with the politician example in the previous section, when the user looks for the available knowledge about that person, an interactive view of the RDF data modeling him is shown.

This way, the user can take profit from the modeling effort and, for instance, be aware of the politician party, that he is a member of the parliament, etc. This interface constitutes a knowledge browser so the link to the parliament can be followed and additional knowledge can be retrieved, for instance a list of all the members of the parliament.

In addition to this recursive navigation of all the domain knowledge, at any browsing step, it is also possible to get all the multimedia content annotated using the concept currently being browsed. This action would bring the user back to the Media Browser. Thanks to this dual browsing experience, the user can navigate through audiovisual content using the *Media Browser* and through the underlying semantic models using the *Knowledge Browser* in a complementary an interweaved way. Moreover, as for the *Media Browser*, the *Knowledge Browser* is also implemented using AJAX so the whole interactive experience can be enjoyed using a web browser.

Conclusions and Future Work

The objective of the S5T project is to integrate different kinds of content through semantically annotated audio transcriptions. Annotations are based on semantic metadata and ontologies. The challenge is then to provide user interfaces that allows user, which are newspaper workers not skilled in Semantic Web tools, interacting with content and annotations in an efficient and effective way.

The main interaction modes are query and browsing. For the former, a query by example tool has been developed. On the other hand, two specialized interfaces have been also developed in order to facilitate media and semantic metadata browsing. The browsing experience is rooted on a metadata graph fragmentation approach. The *Media Browser* is based on this browsing mechanism and allows navigating multimedia content and also semantic annotations.

Keywords in the transcription are linked to the domain ontology concepts that capture their semantics. This links can be followed using the second specialized interface, the *Knowledge Browser*. This second kind of browsing allows users to take profit from the knowledge formalized by the underlying ontologies. Both

browsing mechanisms, the one through media and the one through knowledge, can be combined in order to provide a richer browsing experience.

Future work concentrates on integrating the user interfaces with the rest of the S5T components. The next step is to perform exhaustive user tests of the whole system. However, our long-term goal is to also make the query and browsing tools also available for newspaper end users. On the other hand, we are also exploring an interface for assisted metadata edition that will allow journalists editing semantic annotations and ontologies. Semantic forms are used for this purpose and are also based on the RDF to HTML rendering.

Acknowledgments. The work described in this paper has been partially supported by the Ministerio de Educación y Ciencia under the Programa Nacional de Tecnologías Informáticas, S5T project (TIN2005-06885).

References

- Berners-Lee, T. et. al. Tabulator: Exploring and Analyzing linked data on the Semantic Web. In Proc. of the 3rd Inter. Semantic Web User Interaction Workshop, Athens, USA (2006)
- Bloesch, A.C., Halpin, T.A. ConQuer: a conceptual query language. In Proc. of the Int. Conf. on Conceptual Modeling, LNCS, Vol. 1157. Springer-Verlag (1996) 121-33
- Castells, P., Perdrix, F., Pulido, E., Rico, M., Benjamins, R., Contreras, J., et al. Newspaper Archives on the Semantic Web. In Navarro-Prieto, R., Lorés, J. (eds.): HCI related papers of Interacción 2004, Springer Verlag (2006) 267-276
- Fellbaum, C. WordNet: An Electronic Lexical Database. MIT Press (1998)
- García, R., Gil, R. Improving Human-Semantic Web Interaction: The Rhizomer Experience. CEUR Workshop Proceedings, Vol. 201 (2006) 57-64
- Heath, T., Domingue, J., Shabajee, P. User interaction and uptake challenges to successfully deploying Semantic Web technologies. In Proc. of the 3rd International Semantic Web User Interaction Workshop, Athens, Georgia, USA (2006)
- Hildebrand, M., van Ossenbruggen, J., Hardman, L. /facet: A Browser for Heterogeneous Semantic Web Repositories. In Proc. of the International Semantic Web Conference 2006, LNCS, Vol. 4273. Springer-Verlag (2006) 272-285
- Keramopoulos, E., Pouyioutas, P., Sadler, C. GOQL, A Graphical Query Language for Object-Oriented Database Systems. BITWIT (1997)
- Lieberman, H. (ed.) Your Wish is my Command: Programming by Example. Morgan Kaufmann Publishers. Academic Press, USA (2001)
- Macías, J.A. and Castells, P. Providing End-User Facilities to Simplify Ontology-Driven Web Application Authoring. Interacting with Computers. The Interdisciplinary Journal of Human-Computer Interaction. Elsevier. Volume 20, Issue 1, January (2008) 29-47
- Ozsoyoglu, G.: Example-Based Graphical Database Query Languages. IEEE Computer (1993)
- Shadbolt, N., Berners-Lee, T., Hall, W. The Semantic Web revisited. IEEE Intelligent Systems, Vol. 21, No. 3 (2006) 96-101
- Shneiderman, B. Leonardo's Laptop. MIT Press (2003)
- Tejedor, J., García, R., Fernández, M., López, F., Perdrix, F., Macías, J.A., Gil, R., Oliva, M., Moya, D., Colás, J., Castells, P. Ontology-Based Retrieval of Human Speech. In Proc. of the 6th Inter. Workshop on Web Semantics, WebS'07. IEEE Computer Society Press (2007)
- Zloff, M.M. Query-by-Example: A Data Base Language. IBM Systems Journal, Vol. 16, No. 4 (1977) 324-343