Semantic Skin: from flat textual content to interconnected repositories of semantic data.

Claudio Baldassarre

ABSTRACT

One approach to re-balancing the Digital Divide tends to favor the production of informative content in flat formats, which are easy to distribute and consume. At the same time this approach forbids to deliver the core knowledge pertinent within the content; i.e. it *increases* the Knowledge Divide. In some international organizations[1], informative content distribution to groups in Latin America happens by manually collecting text-based content, then disseminating it via standard mailing lists, or databases copies sent out regularly. Our demo showcases the use of **Semantic Skin** a technology that after semantifying the content submitted in flat formats, provides access to the information via a knowledge layer, which is, however, transparent to the end users. In this context **Semantic Skin** seeks to reduce the Knowledge Divide by bringing in also the pertinent not just surface knowledge to bear.

1. INTRODUCTION

The current version of **Semantic Skin** allows:

1- To create semantic content from flat text: using data-triplification starting from news in static local collections, or from dynamic RSS feeds (a combination of the two cases is also possible). Initial parsing of the content structure is executed[2], next a textual analysis is performed[3] to extract relevant (meta) information[4]. The generation of the knowledge base of news ("news-KB") is obtained by populating an OWL ontology ("news-model") for news (Fig.1), using a triplifier that matches the original data structure on the OWL pattern.

2- To access knowledge-based content: delivered via a front-end web application. This application offers a faceted view of the underlying "news-KB". The current blog site appearance is merely a stylistic choice, while a running instance is always backed by a SPARQL endpoint over the "news-KB". The facets are typically rendered as menu elements[5]: some menus facet the entire "news-KB" (e.g., news Topics, or Provenance); while other menus facet only the content currently visible to the users. The faceting mechanism is also applied tothe "news archive" as a time-based facet of the repository content. All the facets are populated with SPARQL queries over the "news-model" instances in the "news-KB". Each news item is then presented with its summary, title, publication date, and provenance (e.g., permalink). Media (if available) are attached to the news item too.

3- To mash-up remote semantic content repositories: by connecting the endpoints of other running instances of **Semantic Skin**, on the internet. One or more endpoints can be mashed-up within an instance of **Semantic Skin** so to expand the retrieval capabilities from a single search point. The users can select one or more repository from the list of all the available running instances of **Semantic Skin**. The SPARQL/textual queries are distributed to the selected endpoints and the responses are collected. A new widget displays the number of external matches per each remote endpoint.

4- To interconnect remote content repositories: by merging two or more running instances of **Semantic Skin** that are live on the internet [6]. A third, virtual instance, initially blank, is hence generated exposing the merged content as if it was backed by a dedicated 'news-KB'. In a scenario of an organization comprising units, sub-units, divisions, etc., a virtual instance for a sub-unit can behave as an aggregator for multiple divisions, plus adding proprietary or complementary content. The same can apply to units with respect to sub-units.

Semantic Skin is designed, and implemented, by customizing and pipelining (non-)semantic technologies currently available. For this demo **Semantic Skin** is applied to the case of exposing news collections[7].

2. SEMANTIC SKIN AT WORK

[1]The requirements inspiring **Semantic Skin** have been gathered internally to the Food and Agriculture Organization.

[2]News collections are parsed in CSV format or in RSS feeds.

[3]We use the Open Calais web service which allows parsing English, French and Spanish text.

[4]Named entities in the news are identified and categorized (e.g. countries, cities, topics, etc.).

[5]Inspired by Longwell(MIT), and Talis Converter(Talis)

[6]For details visit semanticskin.govirtual.simple-url.com

[7]**Semantic Skin** in its first prototype is based on architectural components, some of which act as placeholders for technologies that can use Semantic Skin as a showcase application.

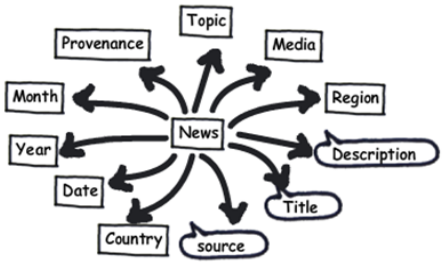

Figure 1: OWL Model for News

To test the functionalities described above and below the reader can visit a running instance[8] of **Semantic Skin**. Alternatively, the approach can apply **Semantic Skin** to one or more RSS feeds the reader is familiar with.[9]. After posting the request to *skin* the RSS feed(s), the process described in (1) takes place at the runtime, creating an endpoint, which is also registered to let other running instances know of its content availability on the internet.

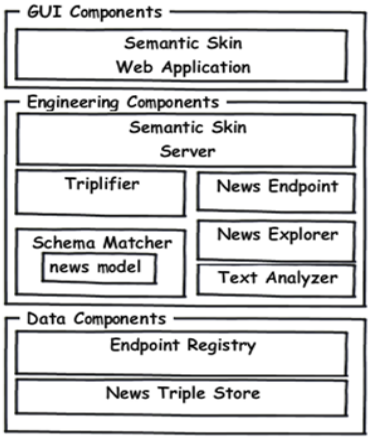

Figure 2: Semantic Skin Components Architecture

In the following some basic user-oriented capabilities of the **Semantically Skinned** blog site are discussed:
i- keyword-driven searching the news: One way users can start searching the collection of news is by keyword-based search. The content of the news is indexed against the "news-KB". Each hit positive to the textual query, returns a reference (i.e., the news instance URI) to an instance in the "news-KB". The interface is populated with the news-content together with additional meta information. The menus faceting the currently displayed news are updated offering more browsing capabilities about the retrieved results. The interface also displays a chronological history of the latest user queries.
ii- interacting with the result of a search: For each news in the result-set the user can: (1)read the summary of the news, (2)reach the original source for the news, (3)watch media if they are attached to the news item.

[8]http://argentina.blogsite.org/: this instance is populated from a local collections of news initially in csv format, and spanish language), then triplified to create the "news-KB".
[9]For details visit semanticskin.goskin.simple-url.com

The user can narrow the result-set by using the faceted browsing: a "click" on a facet instance (i.e., menu items) triggers a SPARQL query to filter-in only the news found to have the facet (i.e., RDF:PROPERTY) value. When the list of news displayed is updated, the menus (i.e., the facets) also updates their content for more browsing. Every action performed on the facets populates the search history chronologically. This comes handy to "jump" back and forth between the executed (SPARQL/Text) queries.
iii- repository exploration: After the "news-KB" is generated, a transformation is applied to create another XML representation to support the explorer of the repository content[10]. Each news is provided with an icon-link as the entry point to the explorer. The user browses the repository content (i.e., the news) along the relationships (RDF:PROPERTY) defined in the "news-model". The news that have the same value for a property (e.g., same provenance, topic, etc.), are connected together in a graph structure.
iii.a- interacting with the News Explorer: The explorer allows the users to reach other news following associations inspired by the relationships in the graph. For example, each item has relationships with: (1)the topics inherent within the news, (2)the press web sites where they have been published (provenance), (3)the countries and the regions interested by the news, (4)the date of publishing.
Each relationship is represented with an arrow connecting two graph nodes (i.e., circles): the news, on one arrow-end, is connected, to the other arrow-end, with one node from the list above. When a node is the center-star of the explorer, all its relationships with other nodes are visible in a circular layout (e.g., all the news with the same provenance). During the news exploration the users can retrieve all the news having a relationship with a selected node (e.g., all the news with the same provenance).
iv- exporting the repository content: Users can export: one single news, the entire result-set of a search, or a collection of news selected individually in several retrievals. When selecting the news during multiple searches, news references (i.e., the news instance URI) are stored in a list of news. When ready, the list is processed to export the "news-model" instantiation for each reference. The news are exported in RDF format serialized with XML syntax.
iv.a- Microformats: The news exposed by **Semantic Skin** are enriched with tags from the hAtom vocabulary. It's hence possible to produce semantic RSS, or perform content extraction via GRDDL enabled tools.

3. CONCLUSIONS

All the technologies mentioned in this document are available as stand-alone products; they are seldom pipelined in a complete knowledge creation and presentation cycle that would span from semantic content production to its consumption. Injecting more semantics collaboratively (to originally flat text) is also a novelty with respect to bigger infrastructures (e.g., Drupal 7) that embed semantic content creation as part of content publishing. **Semantic Skin** being an invisible layer over the content allows not to loose the legacy of: data format, and data retrieval mechanisms. Our technology improves user access to legacy content, exploiting correlation of information in its semantic version.

[10]A customization of http://moritz.stefaner.eu/projects/relation-browser/ is deployed.