



Provided by the author(s) and University of Galway in accordance with publisher policies. Please cite the published version when available.

Title	notitio.us - Semantic Information Discovery, Browsing and Sharing
Author(s)	Gzella, Adam; Kruk, Sebastian Ryszard
Publication Date	2007
Publication Information	Adam Gzella, Sebastian Ryszard Kruk "notitio.us - Semantic Information Discovery, Browsing and Sharing", Proceedings of the Semantic Web Challenge, in conjunction with ISWC 2007, 2007.
Item record	http://hdl.handle.net/10379/552

Downloaded 2024-04-25T10:24:38Z

Some rights reserved. For more information, please see the item record link above.



notitio.us - Semantic Information Discovery, Browsing and Sharing

Adam Gzella and Sebastian Ryszard Kruk

Digital Enterprise Research Institute, National University of Ireland, Galway, Ireland*
<firstname.lastname@deri.org

Abstract. Searching for information on the Internet became much easier when Google started delivering their service. We got used to getting the most important (probably) results on the top of the list. But more and more people realize the limitations of this solution: long list of results - so we never know if the ordering is really the most effective; polluting results ordering with advertisements, and last but not least, almost complete unawareness of meta-data, and what is more, semantics.

The future Web, the one we build as we speak, is the web of highly interconnected meta-data (semantics) and information heavily contributed by the communities of users. Therefore we need more robust, and also more user friendly, solutions to information discovery and sharing. In this article we present notitio.us, a service that allows to aggregate metadata-rich information from various types of social semantic information sources, and provides interesting solutions to further information discovery, browsing, and sharing.

1 Introduction

The future Web, the one we build as we speak, is the web of highly interconnected meta-data (semantics) and information heavily contributed by the communities of users. Therefore we need more robust, and also more user friendly, solutions to information discovery and sharing.

The phenomenon of the Web 2.0 tagging solutions, such as del.icio.us or digg, is based on the fact that everyone can easily contribute to the process of indexing and aggregating information coming from different sources. There are, however some shortcomings, in contrary to search engines like Google, the content of the sources being indexed is not stored together with users' annotations. Usually, the annotations themselves, are mere list of tags, and users are forced to use only simple keywords based search and filtering.

The Web, however, becomes more and more rich in the semantics. These could be exploited in the discovery and browsing process. Moreover, the community process does not have to be limited only to the information indexing, but could also be used to enhance the user experience in knowledge sharing.

* This material is based upon works supported by Enterprise Ireland under Grant No. ILP/05/203. Authors thank Stefan Decker and Bill McDaniel for all their help, and all members of the Corrib community for fruitful discussions on this project.

In this article we present <http://notitio.us>, a service that allows to aggregate metadata-rich information from various types of social semantic information sources (see Sec 2). Notitio.us allows users to easily discover and share their knowledge (see Sec. 3). It also provides an interesting solution to further information browsing, using either faceted navigation (see Sec. 5) or tags-based filtering (see Sec. 4).

Information aggregated from various sources, like wikis, blogs, fora, and digital libraries, is exposed in either pure RDF or LOM [3] with a set of REST services delivered by notitio.us. Users contribute to the growth of the knowledge base while bookmarking and classifying interesting sites. The semantic annotations of the indexed resources can be exploited with the faceted navigation, or used by the recommendation engine (part of the SSCF component - see Sec. 3) to suggest interesting information bookmarked by other members of the community.

In the following sections we will describe each of four main modules, of which the notitio.us service consists. More information about this service can be found at <http://wiki.notitio.us>. We will conclude description of how this modules work together in the context of the Semantic Web Challenge.

2 IKHarvester - Informal Knowledge Harvesting (IKH)

IKHarvester (Informal Knowledge Harvester) [2] is a web service that allows to harvest meta-data, and provides it for various information management frameworks, e.g., e-Learning. IKHarvester supports the Semantic Web principles such as rich descriptions of resources; with rich semantics the content of web pages can be understood not only *with* machines but also *by* machines.

IKHarvester exposes all of its functionality with REST-based SOA¹. It allows to access and manipulate data with ease.

Data harvesting IKHarvester captures RDF data from Social Semantic Information Sources (SSIS) [7]. The current version works with semantic blogs, semantic wikis, and JeromeDL (the Social Semantic Digital Library) [5].

IKHarvester looks for RDF documents related to the given resource, which is indicated by a special LINK HTML entry. Data captured from online communities, like blogs, wikis, bulletin boards, can be described with SIOC ontology, whereas JeromeDL and MarcOnt ontologies are employed for describing bibliographic resources. Besides reading pure RDF data, IKHarvester uses Microformats which allow embedding RDF into HTML documents. Moreover, IKHarvester is capable of creating RDF descriptions for non-semantic information sources such as Wikipedia. For that reason, it scrapes the HTML code of an article and collects a title, external links, *see also* links and references. The RDF document which was read through LINK tag, or scrapped from the content is saved in the informal knowledge repository.

¹ IKHarvester SOA specification: <http://wiki.didaskon.corrib.org/IKHarvester/REST>

Each time a user invokes harvesting process, he/she can additionally tag the harvested document. System provides a list of tag that was used to annotated the given document before.

Accessing data Once the informal knowledge repository is filled with data, it is exposed for other services. System stores information in RDF, so with a little effort data can be provided in almost any format. So far IKHarvester allows to export information in LOM (Learning Object Metadata) [3]. It is a format used by Learning Management Systems (LMSs). Therefore IKHarvester provides informal learning material in the form of the formal Learning Object.

Extensibility Current version of IKHarvester operates on three types of resources: SIOC-enabled blogs and fora, wikis that use MediaWiki engine, and JeromeDL. However, there are other types of web pages that could be captured. Therefore, IKHarvester architecture allows adding new extensions for those websites. We hope that more and more extensions will be provided in the future to cover more sources of informal knowledge, which do not expose metadata explicitly.

3 Social Semantic Collaborative Filtering (SSCF)

The main aim of the Social Semantic Collaborative Filtering (SSCF) is to allow users to save the knowledge and share it with others.

Users maintain their private collections of bookmarks to the interesting, valuable resources. In other words, each user gathers, filters, and organizes a small part of knowledge. What is important, SSCF allows a user to share this knowledge with others in the social network; one can easily import friends' bookmarks and utilize their expertise and experience in specific domains of knowledge.

In SSCF users collect bookmarks and store them in special directories; each directory is semantically annotated using popular taxonomies, such as WordNet, DMOZ, or DDC, or other. These annotations can be used to determine the content of the directory or to find the correct one.

Users can include information from friends by importing their directories into her/his own. The knowledge is based on the bookmarks of interesting and valuable books, articles or other materials. SSCF can be used to bookmark various types of resources, e.g., from digital libraries, community-enabled sites (blogs, fora), or just standard web resources.

Another important aspect is the security in SSCF. System allows users to set fine-grained access rights for every directory; access control is based on the distance and the friendship level between friends in the social network.

People sharing similar interests can have problems in meeting each other in the social network due to too high degree of separation between them. SSCF is equipped with preliminary version of a suggestion engine, which goal is to overcome this problem. It is a prolog-based tool which tries to find similar persons in the network and suggest each other SSCF directories. When user finds the suggestion valuable he/she can import the directory and extend the friends list with user that is owner of such a directory.

SIOC integration In the current Web, blogs become more and more popular. There are many different types of blogs; sometimes, they are published by a person with a good expertise in a certain domain. A lot of knowledge is also delivered through the Web fora; the discussions are topic-oriented. They, very often, contain solution to problems, or point to other interesting posts, which add valuable views in to the debate. Such sources are rich in knowledge; therefore, it is crucial to use their potential. We have incorporated SIOC [1] into SSCF model and SSCF bookmarks interface. There is a special directory dedicated for storing SIOC data in a private bookshelf. This catalogue can maintain three types of SIOC concepts; users can bookmark posts, or whole fora or sites. For each resource, it is possible to browse the content. The SIOC-specific resources behave just like classic SSCF ones; a user can copy a SIOC entry and paste it into another SSCF directory. This way, a standard knowledge repository is enriched with community based content.

del.icio.us integration del.icio.us is one of the first and one of the most popular Web2.0 tagging service. We have decided to provide our users with possibility to import del.icio.us bookmarks to SSCF, so they will keep already gathered information.

Both del.icio.us and SSCF was build to store and provide bookmarks to interesting and valuable web resources, but each system does it in different way. del.icio.us use tags to annotate each resource; in SSCF resource are annotated with specific position in hierarchy of well annotated of directories. To adapt del.icio.us bookmarks we create directories from tags and bundles and then we are clustering similar ones together. Such structure is filled up with bookmarks.

Model evaluation The model evaluation of the SSCF [4] revealed that each user is able to find (on average) the best quality of information provided by other users within the subgraph of a social network bounded by 6 degrees of separation. These results proved that the constructed social network model corresponds to the small world phenomena. Hence the assumption underlying the social collaborative filtering has been fulfilled. It is possible to find an expert within the small social network neighborhood.

4 TagsTreeMaps (TTM)

TagsTreeMaps allow to represent flat set of tags as a tree of clustered categories, using treemaps algorithm. It allows to filter out tags based on name of the tags (and sub-tags) and frequency of using tags. Users can select tags and create summary or conjunctive filter on current set of results. TTM implements zooming paradigm, and users can easily move vertically on the tree of clustered tags.

TTM solves one of the fundamental problems of tagging space; when trying to browse through a large space of tagged information, it is very hard to get a hold on a large number of tags used. Currently used solutions like tag clouds present static approach to render and differentiate tags. Eventually, user has

problem when trying to find appropriate tags to filter the information. Very often the tags cannot be filtered by any means and they are always represented as a flat structure.

TagsTreeMaps delivers pluggable architecture for: (1) filtering of tags, e.g., min/max usage, filtering by tag's name; (2) preprocessing tags, e.g., clustering tags based on predefined bundles of tags or string similarity distances; (3) post-processing treemap, e.g., appropriate coloring tags so that they are distinctive and much easier to read. TagsTreeMaps performs zoom in/out actions (multi-level) to a selected group of tags, clusters, bundles. Additionally, user can choose between union mode (at least one of the tags in the selected view should be used in the information filtered) or conjunctive mode (all tags must be used) – as opposed to current solutions where only conjunctive mode is delivered.

Within notitio.us TTM allows to browse seamlessly information provided by both notitio.us and del.icio.us, provided a user has account in these systems. But the list of providers can be easily extended to other systems.

5 MultiBeeBrowse (MBB)

MultiBeeBrowse allows to browse unstructured metadata represented as an RDF graph. System is build according to SOA (Service Oriented Architecture) paradigm, coupled with AJAX-based user interface.

REST based services All the services build for MBB are RESTful. An argument for REST, in the context of the MultiBeeBrowse service, is that GET action defines an idempotent request, i.e., subsequent calls of the same URL should return the same results. This can ensure that user will get the same results, each time given URL is called. In MBB it is vital for and handling history of results. Our goal was also to construct a meaningful URL representing single browse operations, as well as, whole chain of operations building up a browsing query. This would allow advanced users to quickly construct their queries, directly in the web browser address field.

We have identified following types of services that should be delivered by our SOA: (1) browsing services (access to a resource, search, filter, browse, similar, combine) (2) context and history management services; (3) meta-services, which provides access to statistical information, and allows to format response in desired metadata language format.

Each service is specified using BNF notation²; it is enough, since most of the services except for the context services provide only GET method implementation. Each BNF specification has been translated into a regular expression. All services has been grouped in a hierarchical structure.

Browsing services, deliver the primary functionality of the MultiBeeBrowse component. It consist of: (1) Access resource service, to load metadata about a resource with given URL for further browsing; (2) Search services - keywords

² <http://wiki.s3b.corrib.org/MBB/SOA>

and advanced search, natural language query templates [8], and direct RDF query service; (3) Filter service to specify selection filter; (4) Similar service to find resource similar to those in given set; (5) Related service to find resource that are related to the given ones with given property; (6) Combination service, which performs four operations: conjunction, sum, difference, binding, on two given sets of results.

The information on the context of browsing is kept in the RDF storage according to a simple ontology. This ontology defines a Browsing Context as a set of current browsing queries. With context services, users can traverse current browsing contexts, or retrieve all their browsing contexts with the given call.

Meta-services allow to render results in one of RDF serializations, or in one of feed (RSS, Atom) formats. For adaptable user interface purposes a special meta-service generates statistics on properties and values a user can select from. Other generates a list of most frequently used concepts, renders the definition of current chain of browsing operations in HTML or XML or generates a unique ID for given browsing query.

MBB User Interface Based on these two goals: handling many paths of back and forth refinements, and access to a structured history of operations, we have identified 4 views of browsing context, which allow user to access effortlessly each of aforementioned features. (1) Basic browsing view provides access to all browsing operations with a typical search and browsing user interface. To further enhance usability of our solution, we have extended the query building part with suggestions of properties and values, and results rendering part with an in-site browsing menu; (2) Structured history view allows users to view their current results in the context of previous and following (if any) operations. This view is almost the same as the basic browsing view, with one difference that users see 6 slots with previous operations, and another 6 slots for further browsing; (3) HoneycombTM view presents users a comprehensive overview of their current browsing context. Each browsing query is represented with a hexagon lozenge in a 3D visualization; some edges between hexagons represent browsing operations that were added to the chain of operation to create a new browsing query. With this view users can get a quick overview on their browsing session or jump to a selected browsing query in the current context. This view also allows to perform Combine operation or perform any browsing operation, which was allowed in previous views. It is done by clicking on hexagon edges; (4) Life-long history view presents all previous sessions in which the given query was invoked. It allows a user to quickly move in time to some browsing context, review refinements invoked after the given browsing query. User can even jump back to that context and continue browsing.

Evaluation MultiBeeBrowse has been evaluated [6] against two other similar solutions: BrowseRDF [9] and Longwell³. It outperformed, according to a group of test subjects, BrowseRDF, and came close in those features which were also supported by Longwell. Since MBB delivers unique features, such as collaborative

³ <http://simile.mit.edu/wiki/Longwell>

browsing, adaptive user interface, and open SOA for delivering new user interfaces, it turned out to be an interesting solution for browsing on unstructured metadata.

6 notitio.us modules working together

notitio.us is a service that connects all the aforementioned technologies to provide users with full featured Semantic Web application. Each service plays its specific role and they all work together to achieve most demanding requests.

Providing knowledge (IKH and SSCF) It can be hard to motivate users to create informal knowledge repository using IKHarvester. In the same time users are keen to save and share interesting bookmarks to valuable web resources. We have used that fact and every time user is bookmarking some resource, system calls IKH which tries to harvest the resource behind the bookmark. As a result system is collecting the knowledge, which is already filtered, when somebody found valuable.

Utilizing and browsing knowledge (MBB, TTM, SSCF) In notitio.us users can search and browse information in various ways. First of all they have easy access to the bookmarks and bookmarks of their friends and others from social network. It is a good way to find some valuable information and suggestion process helps discovering potential domain expert.

TTM allows to easy browse through large set of tags and get desired information. While harvesting pages with IKHarvester users can tag the information. With TTM they could easily filter through a set of IKH tags and find desired resource.

MBB is a universal faceted browser that can work on unstructured data. In notitio.us MBB features, i.e., context and full history of search process and unlimited refinement possibilities have been extended with collaborative browsing.

Collaborative browsing solution is based on MBB REST URIs in which all the actions in the system, facets dependency and the context of the search are saved. By invoking such a URL users can retrace all their browsing steps and get desired results. It is also possible to refine the query if the result is insufficient or the search objective has changed. In notitio.us users can use SSCF to simply add such search process to their bookmarks. To make it easier under each search field we have placed a link which opens a pop-up window with users bookmarks' structure. When MBB search is bookmarked, it can be used and shared just like an ordinary SSCF resource.

Exporting knowledge (IKH, MBB) Getting semantic information from notitio.us is possible with REST-based SOAs of IKH and MBB components. IKH was build to provide data in many different formats and is easy to extend. With IKH information stored in the repository can be also represented using LOM (Learning Objects Metadata); hence, other Learning Management Systems can build upon the information aggregated by notitio.us.

Even though MBB has been build as a browsing solution, its open SOA specification allows other services to query and access information stored in the repository. Results of each search and browsing query can be exported to one of RDF formats (XML, N3, N-Triples, Turtle).

7 Conclusions

In this article we have presented notitio.us, a service that allows to aggregate metadata-rich information from various types of social semantic information sources, and provides interesting solutions to further information discovery, browsing, and sharing. Components of notitio.us has been described and evaluated separately and are currently under the peer review process in different conferences.

References

1. J. G. Breslin, A. Harth, U. Bojars, and S. Decker. Towards semantically-interlinked online communities. In *2nd European Semantic Web Conference 2005*, 2005.
2. J. Dobrzanski, T. Nagle, E. Curry, A. Gzella, and S. R. Kruk. Ikhvester – informal elearning with semantic web harvesting. Technical report, DERI eLite Project Deliverable 1.6.08, <http://library.deri.ie/resource/kvtRQS90>, 2007.
3. IEEE. Draft standard for learning object metadata. Technical report, Institute of Electrical and Electronics Engineers, Inc., http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf, 2002.
4. S. R. Kruk and S. Decker. Social Semantic Collaborative Filtering with FOAF-Realm. In *Semantic Desktop Workshop, ISWC 2005*, 2005.
5. S. R. Kruk, S. Decker, and L. Zieborak. JeromeDL - Adding Semantic Web Technologies to Digital Libraries. In *Proceedings of DEXA'2005 Conference*, 2005.
6. S. R. Kruk, A. Gzella, F. Czaja, W. Bultrowicz, and E. Kruk. Multibebrowse – accessible browsing on unstructured metadata. In *proceedings of ODBASE2007*, 2007.
7. S. R. Kruk, A. Gzella, J. Dobrzański, T. Woroniecki, and B. McDaniel. E-learning on the social semantic information sources. In *Proceedings of EC-TEL'2007*, 2007.
8. S. R. Kruk, K. Samp, C. O'Nuallain, B. Davis, B. McDaniel, and S. Grzonkowski. Search interface based on natural language query templates. In *Proceedings of IADIS International Conference WWW/Internet 2006*, 2006.
9. E. Oren, R. Delbru, and S. Decker. Extending faceted navigation for rdf data. In *Proceedings of ISWC'2006*, 2006.