Journal of Universal Computer Science, vol. 16, no. 21 (2010), 3278-3298
submitted: 15/5/09, accepted: 24/11/10, appeared: 1/12/10 © J.UCS

Integrating Personal Web Data through Semantically Enhanced Web Portal

Lidia Rovan, Tomislav Jagušt and Mirta Baranović
(Department of Applied Computing
University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
{lidia.rovan, tomislav.jagust, mirta.baranovic}@fer.hr)

Abstract: Currently, the World Wide Web is mostly composed of isolated and loosely connected "data islands". Connecting them together and retrieving only the information that is of interest to the user is the common Web usage process. Creating infrastructure that would support automation of that process by aggregating and integrating Web data in accordance to user's personal preferences would greatly improve today's Web usage. A significant part of Web data is available only through the login and password protected applications. As that data is very important for the usefulness of described process, proposed infrastructure needs to support authorized access to user's personal data. In this paper we propose a semantically enhanced Web portal that presents unique personalized user's entry to the domain-specific Web information. We also propose an identity management system that supports authorized access to the protected Web data. To verify the proposed solution, we have built Sweb - a semantically enhanced Web portal that uses proposed identity management system.

Keywords: Web portal, Semantic Web, OpenID, OAuth, Web architecture, personalization
Categories: D.2.11, H.3.4

1 Introduction

Nowadays, the World Wide Web contains enormous amounts of data scattered across numerous Web sites and applications. The process of finding, accessing, collecting and combining that information, in order to create new conclusions, is getting more and more time-demanding. Automation of that process will, we believe, bring significant improvements in the Web usage. However, in order to automate user's common Web usage, two major challenges must be resolved: a *personalization* and transparent *Web data integration*.

We propose realization of mentioned process by developing a semantically enhanced Web portal because we believe that Semantic Web technologies could bring significant improvement on the realization of the main implementation issues. The main purpose of Semantic Web technologies is to bridge data of different structure and format, so they are commonly used in data integration implementations. Moreover, the latest development directions in the field of Semantic Web are focused on the "Web of data" or "Linked data" principles [Berners-Lee, 10]. Additionally, improvements that Semantic Web technologies could bring to the personalization problem are also recognized [Vuljanić, 10].

Personalization is defined as the ability to provide content and services tailored to individuals based on knowledge about their preferences and behaviour [Liang, 10].

Therefore, it can be assumed that a personalization system delivers better results when having at disposal a broader set of user's personal data, usually stored in login and password protected Web applications.

In general, the available Web data, depending on the access method, can be divided into two subsets:

- publicly available data
- data accessed through the access control system

While there are standardized Semantic Web protocols for accessing publicly available Web data [Clark, 09], standard that defines how to implement authorized access and retrieval of the protected data between remote applications/services is still not proposed. In this paper, as a scientific contribution, we propose a solution for the problem of authorized access to protected data on the Web using the Semantic Web technologies. We developed a prototype of a semantically enhanced Web portal that uses the proposed solution for achieving authorized access and could represent a comprehensive solution to a unified data access on the Web.

This paper is organized as follows. We start out, in Section 2 by elaborating the idea of semantically enhanced Web portal, explaining the need of architectural requirements the portal should meet and emphasizing the main technical issues it deals with. In Section 3 we propose identity management system that supports authorized access to the protected Web data. Section 4 presents the developed Web portal prototype. In Sections 5 and 6 we analyze and discuss possible solutions of technical issues: personalization, customization, data aggregation and data integration, respectively. Section 7 gives an architecture overview of the portal we developed. Finally, we conclude in Section 8 with a brief summary and a short outlook to the future work.

2 Semantically Enhanced Web portal

The idea of a Web portal is to collect information from different sources and create a single point of access to data, expertise and applications. Hence, the portals are designed as environment of data integration and a personalized display. In other words, portals are designed as an environment that provides support in performing user's everyday tasks from a particular domain. Today, the different applications are used to fulfil individual needs and interests of a particular user. For example, Flickr[1] is used to organize pictures, Facebook[2] as a social network tool, Google Calendar[3] as an organizer, etc. The functionality of each application is developed in a way that is the most convenient to the user, which, as a side-effect, causes the fragmentation of personal data on the Web. To efficiently perform frequent daily tasks, there is a need to integrate portal domain data and user's personal data stored in various Web applications. For example, one possible scenario is: the Web portal based on user's preferences offers a personalized list of events that the user might find interesting, user finds one such interesting event and decides to integrate it into her calendar

[1] [1] www.flickr.com
[2] [2] www.facebook.com
[3] [3] www.google.com/calendar

located in another application, at the same time wants to identify potential collisions in the schedule, and notify her friends using the contacts list from a third application.

Thus, the portal that suits the needs of today's typical user presents integrated data from a chosen domain and connects it with user's personal information stored in various applications. Specifically, the portal integrates:

- its own data
- publicly available information on the Web
- various data services accessed through the access control systems

Main functionalities that Web portal has to include are: data aggregation, data integration and data personalization. Although the technical realization of the main portal functionalities can be significantly improved with the usage of Semantic Web technologies, our observation was that these technologies are mostly used in the development of non critical applications, with no need to provide a completeness of solution. Lack of development of architecturally complete solutions resulted in a lack of technologies for authenticated access to the protected Web data. Even though, the recently formed Linked Data principles have made a huge impact on the standardization of the methods used in the Semantic Web data integration [Bizer, 09a, Bizer, 09b], the problem of including protected data in the data integration process has been left intact.

In order to meet the demands of today's most successful Web applications, we set the following architectural requirements that the Web portal developed using the technologies we propose in this paper additionally has to satisfy: [Murugesan, 07]

a) maintainability (extensibility, adaptability and simplicity)
b) security
c) usability (data coherence and high user-interactivity)
d) scalability
e) robustness and reliability
f) interoperability

Software architecture is defined as a configuration of architectural elements constrained in their relationships in order to achieve a desired set of architectural properties [Perry, 92]. We have performed analysis of the available Semantic Web recommendations and development tools and made adequate choices for each of technical problems. We have tried to choose the ones that best meet given architectural requirements and follow World Wide Web Consortium (W3C) recommendations, so our solution would be in accordance with established standards.

We discovered that by using the existing technologies only, our solution does not meet all the set architectural requirements, Therefore, as proposed solution to the perceived problem, we developed a system for identity management in the semantic Web environment and here we propose its architecture and explain the way it is used in semantically enhanced Web portal. During development of the prototype, our hypotheses were verified and architectural trade-offs, along with needed discrepancies from standardization, were pointed out.

3 Achieving authorized access to personal Web data

The process of retrieving data from the remote location through the access control system includes user identification, authentication of remote applications, and finally, the data retrieval itself. The analysis of the research results in the field of protected data access [Grzonkowski, 05; Kruk, 04; Pashalidis, 03; Samar, 99; Suriadi, 09] has shown that there was no adequate infrastructure and methods that could meet the objectives set out in this research. However, by examining the available solutions and achievements in the field of identification [OpenID, 09], authorization [Atwood, 09], methods of storing user's personal data [Brickley, 08] and the mechanism of data access [Clark, 09], we discovered that combined usage of research results in these areas could provide a desired solution. We propose a system for identity management on the Web and an expansion of the SPARQL protocol (SPROT) [Clark, 09], that used together with a semantically enhanced Web portal, allows a secure Web data integration. Below is a detailed explanation of the proposed solution.

3.1 Identity management system

An identity management system is thought as both an authentication system and attribute management system. There are two distinct approaches to identity management: “centralized identity management” and “federated identity management” [Miyata, 06]. In this paper we propose a centralized identity management system. We argue that the identity management system, for successful operation in the Semantic Web environment, must have the following characteristics:

1. enabling a single sign-on
    - only one identifier / authenticator pair (eg, user name and password) for all services
    - single user login action for all available services/applications
2. user profile understandable to computers and stored in a way that allows easy sharing
3. ensuring the safety, privacy and data protection

During the implementation of system that accesses and retrieves protected data, we discovered that the main technical problem was to achieve the secure access procedure, while preserving architectural requirements of usability and interoperability. Usability here implies a transparent access to all data sources, or the process of identifying, collecting and integrating data without explicit user involvement. Single sign-on is a mechanism that provides required behaviour. In order to allow interoperability, we proposed semantic data storage and developed protocols for the secure exchange of semantic content.

The Table 1 shows a summary of all decisions made in defining and creating an identity management system for the Web environment. The reasons for every decision are provided explaining the influence of that decision on the semantically enhanced Web application architecture that uses the proposed identity management system. Every decision is explained in detail in the text below.

decision	**reason – impact on the semantic Web applications architecture**
separation of identity management system	easier application maintenance
single sign-on	usability
one identifier / authenticator pair	interoperability
single user login action	usability
semantic user profile	interoperability
SPROT protocol extension	maintainability scalability interoperability
access control system	security privacy

Table 1: Reasons for decisions that were made during the creation of the identity management system

3.2 Single sign-on

The solution of described problem requires access to protected data stored in various applications. To avoid continuous repetition of the sign-in procedure in each application individually, it is necessary to implement a single sign-on. A single sign-on feature means that user can access multiple applications/services without the need for a separate login to each site. So, in this way, the access to all protected data is allowed, while preserving the ease of use.

During the time, number of applications/services that average user accesses on daily basis has grown. Remembering passwords and managing identities for that number of services is cumbersome for the users. The problem of user profile maintenance can be viewed from two sides, the user and the developer side. The identity maintenance costs become higher over the time. Frequent problems are forgotten passwords, incorrect and inconsistent data, the need for communication with other services, and most important - security [Samar, 99]. These reasons are in favour of the decision that the identity maintenance should be separated from the applications themselves, because such a solution would obviously reduce the maintenance cost. Easy maintenance is one of the requirements on the architecture of Semantic Web applications that were set out in this study. Isolation of identity management system from the applications means that users must have only one identifier / authenticator pair for multiple applications.

3.2.1 Single identifier and authenticator

For the identity management system we chose OpenID [OpenID, 09] as solution to the first requirement stated before (only one pair of identifier and authenticator). OpenID is an open, decentralized, free framework for user-centric digital identity; that is a decentralized mechanism for a single sign-on. There are two reasons why this mechanism was chosen:

- as an identifier OpenID uses Uniform Resource Identifier (URI), which means that it fits into the Semantic Web principles
- it is globally accepted (a number of existing applications use it)

It is necessary to further explain what exactly means to use only one identifier / authenticator pair. The user can choose any preferable service as her OpenID service provider (eg. getopenid.com, myopenid.com, etc ...), and as such it is a decentralized mechanism. Also, the user can create and maintain an unlimited number of identities, if there is a need for that. So, if necessary, the user can have multiple URIs, mutually independent. To achieve a single sign-on, applications must support the OpenID sign-on mechanism. Thus, the OpenID URI can be treated as a user's global unique identifier. The introduction of the global identifier seems to be in a contradiction with the basic principles of the Semantic Web, since it introduces a form of centralization. Introducing centralization is essential for achieving the integration of user's private information, because if it was allowed for users to have different URI-s in various applications, integration of user data would require the use of some form of URI equalization. Currently, all the approaches for discovering and setting links between semantic data sources are based on the similarity metrics [Hausenblas, 08; Hassanzadeh, 09; Raimond, 08; Volz, 09; Wang, 06]. Implementing integration according to the Linked data principles [Bizer, 09b], hence searching for patterns or using similarity metrics, causes the following problems: the accuracy is not guaranteed, the use of similarity metrics requires that an application has access to protected data over which the inference would be done what violates the privacy of users (eg. the comparison of name, mail address, etc.). Consequently, without the mediation on the higher level (e.g. between data owners) automatic personal data integration is impossible. Therefore, OpenID URI can be regarded as the global Web identifier. Further, when appropriately generated it does not have to be self-explanatory, thus not revealing any personal information.

We note here that OpenID system as an authenticator uses password, so vulnerabilities of using passwords as authenticator [Notoatmodjo, 07] should be taken under consideration.

3.2.2 User profile

The choice of technologies for storing a user profile has a strong influence on the architectural requirement to support interoperability. That condition can be reformulated in the following statement: *1) user profile must be understandable by computers*, and *2) multiple services should be allowed to access and use it*.

A large number of researches propose usage of the FOAF (Friend-of-a-friend) vocabulary [Brickley, 09] for describing profile data [Ankolekar, 06; Grzonkowski, 05; Kruk, 04]. It is a RDF(S) (Resource Description Framework (Schema)) vocabulary, so the requirement that computers must understand the data is fulfilled.

Possibility of integrating FOAF vocabulary and the OpenID technology is also recognized, so the latest FOAF specification includes an OpenID support [Brickley, 09]. Since FOAF has a limited set of properties, for the description of user profiles a more general solution is proposed. It is also based on FOAF, but it can use any number of RDF vocabularies.

In order to satisfy the second requirement, collaborative use of profile data by remote services, standardization of data access becomes an imperative. The Web portal must be able to access various service providers (eg Google, Verisign, etc..), depending which one was chosen as the user's OpenID provider. In case when such access is standardized, developing and maintaining costs of Web portal are much lower. Since the profile is written in RDF and stored at a remote location, a natural choice for data access is the SPROT protocol and the SPARQL access point.

3.2.3 User profile data access

The collaborative use of user's profile rises, besides technological and legal questions. Here, we emphasize the personal information protection. Users want different subsets of data to be visible to different services (see Figure 1). The focus of existing solutions for identity management on the Internet is on the identification and authentication while this study emphasizes the authorization of data access. To provide appropriate level of security, including data protection, access control analogous to the one in relational database systems has to be implemented.

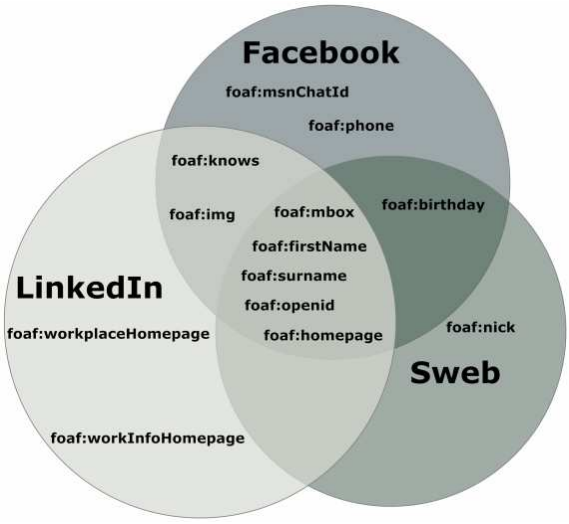

Figure 1: User profile service permissions

Appearance of OpenID 2.0 specification made storage and exchange of arbitrary attributes possible [Fitzpatrick, 09]. That solution has two major disadvantages: *1) the same information is available to all accessing services* and *2) attributes do not have a well defined meaning*. Using RDF to store the profile data eliminates the second problem. Solving access control issue is a much bigger challenge. As an access

technology we propose the SPROT protocol. Our approach is made in accordance to the best practices from the database field, based on the belief that in the future RDF triple stores will be used as backends for applications in analogy to existing relational databases [Dietzold, 06]. Performance of triple stores is constantly increasing, so in the near future data storage alone should no longer be an issue [Bizer, 09c]. SPARQL endpoint (REST Web service) is a chosen data access method mainly because REST based design is both a fundamental enabling technology of Web 2.0 and a natural fit for Semantic Web operations [Battle, 08].

Access control for the proposed identity management system includes the ability to limit access to services. Additionally it can allow access (and the type of access) for each service only to the specific data. The Figure 2 shows the interface through which the user assigns permissions in the identity management system implemented as a part of this research. The smallest granularity is granting access on the triplet (statement). Based on the set of permissions over the triples, semantic repository can perform access control. Access control between the accessing service and the SPARQL access point is carried out by OAuth technology.

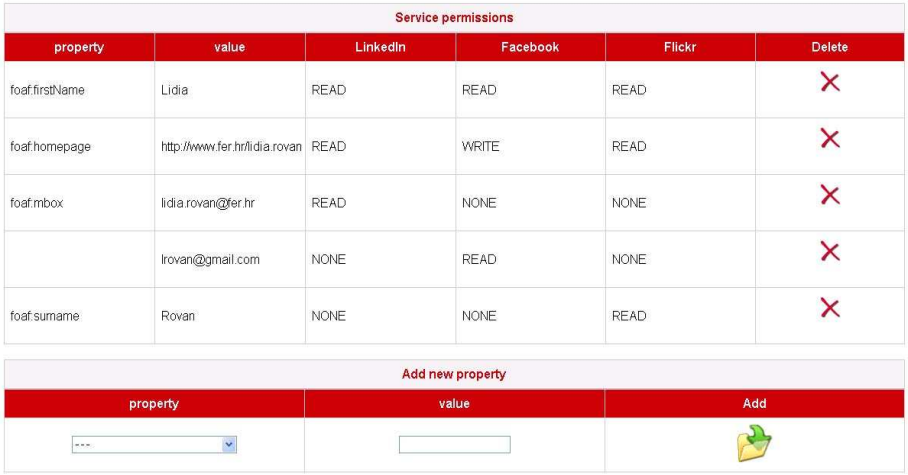

Figure 2: Triple permissions – user interface

OAuth [Atwood, 09] is an open protocol that enables secure API authorization and is now imposed as a standard access control mechanism to the services that access user's private data. The proposed solution for authorized access to user data stored in the identity management system consists of the following basic components (see Figure 3):

1. hybrid OpenID/OAuth service provider
2. access control framework
3. SPARQL Protocol for RDF (SPROT) protocol extension

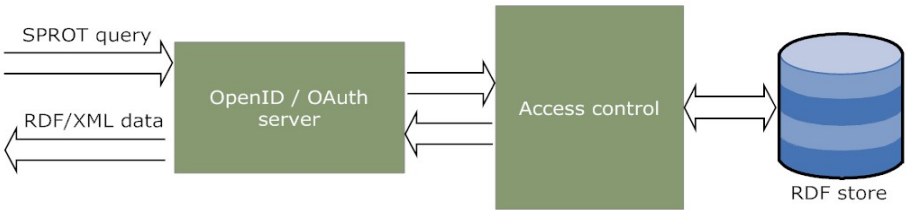

Figure 3: Identity management and single sign-on system

3.2.4 Hybrid OpenID/OAuth service provider

OpenID is used in conjunction with OAuth to delegate access to the service required by the consumer. Here, a final destination is a service which provides user's profile data. To enable the data retrieval, user has to be authenticated and access to consumer application has to be authorized by the user. Authorization rights could be set in advance trough the preferences page or during the acquisition process. Data flow is shown in figure (see Figure 4).

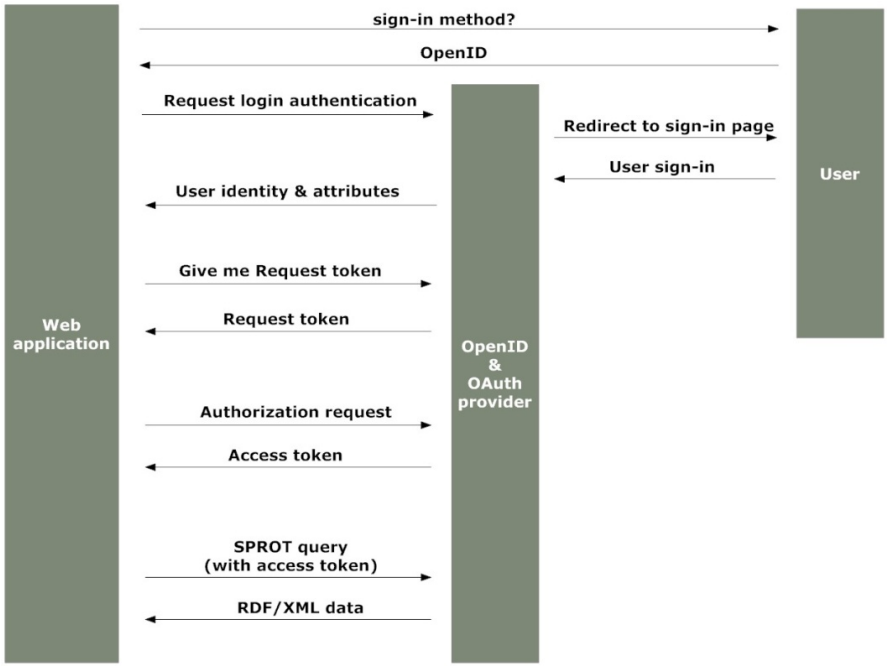

Figure 4: Single sign-on data flow

3.2.5 SPARQL Protocol for RDF (SPROT) protocol extension

During the development a lot of care was devoted to the usability of proposed system by the developer's side. For that reason the entire process on the remote service side is

performed by a single SPROT query. In order to implement such approach, we propose SPROT protocol extension, in the form of additional parameters that contain the necessary data for OpenID / OAuth communication.

SPROT is a method of performing queries through SPARQL endpoint. SPARQL URI consists of three parts: 1) SPARQL endpoint URL, 2) the graphs to be queried against and 3) the query itself.

The SPARQL URI consisting of these three parts looks like this:

```
GET/endpoint/profile/                                    HTTP/1.1
Host:                                  http://www.sWeb.zpr.fer.hr
Content-Type:                                application/atom+xml
query="SELECT%20DISTINCT%20%3Fname%20%3Fmbox%0D%0AWHERE%20%
7B%20%3Fx%20foaf%3Aname%20%3Fname%3B%20%09%20%0D%0Afoaf%3Am
box%20%3Fmbox%7D%0D%0A"
```

Figure 5: SARQL URI example

We propose an extension of the SPARQL URI to include parameters needed for OAuth authentication. Parameters are included in the Authorization header following OAuth 1.0 specification [Atwood, 09] for accessing protected resources. Figure 6 shows an example of the extended SPARQL URI:

```
GET/endpoint/profile/                                    HTTP/1.1
Host:                                  http://www.sWeb.zpr.fer.hr
Content-Type:                                application/atom+xml
query="SELECT%20DISTINCT%20%3Fname%20%3Fmbox%0D%0AWHERE%20%
7B%20%3Fx%20foaf%3Aname%20%3Fname%3B%20%09%20%0D%0Afoaf%3Am
box%20%3Fmbox%7D%0D%0A"
Authorization:OAuth
OAuth_token="1%2Fab3cd9j4ks73hf7g",
OAuth_signature_method="HMAC-SHA1",
OAuth_signature="WyKGTLsxOfTLcDIVwH5hHeqzpwI%3D",
OAuth_consumer_key="sWeb",
OAuth_timestamp="1231956529",
OAuth_nonce="57451704142536",
OAuth_version="1.0"
```

Figure 6: Extended SPARQL URI example

3.2.6 Access control framework

After the successful OAuth authorization process, consumer gains the access token which presents a key to the data access services (see Figure 4). Application uses that token for all further communication to the provider. Since every access token is specific to only one user and application, it is easy for provider to read out the needed information: user and consumer application. That two parameters are passed to the access control tool and proper query filtering is performed. Before consumer request is forwarded to the access control framework, OAuth provider confirms validity of the

access token. Query results are returned to the consumer in accordance to the SPROT specification.

Triple stores access control is crucial and results in that area could trigger greater acceptance of the Semantic Web technologies. From an architectural point of view, at the moment, application security is the weakest part of the Semantic Web. The lack of open access control solutions forced us to build our own light-weight system as a proof of our ideas. For each application that is registered as a user profile consumer we construct two RDF models: one containing only triplets with the read-only permission and the other containing triplets with the modify permission. As a result, application has access only to the subset of user profile data that matches to the set policy.

Although there are perceivable results in the aceess control field [Bizer, 09a; Dietzold, 06; Manjunath, 09; Jain, 06; Reddivari, 07], without a more serious implementation we will not have the objective feedback. For example, none of the approaches provides user interface for policy management which affects usability. But what seems to be the most discouraging in this area is discrepancy from standardization, importance of which we already emphasized.

4 Sweb

Sweb is a semantically enhanced Web portal intended for the student population. The services it provides can be classified into two categories: entertainment information and personal services. (see Figure 7) The entertainment information is gathered using RSS/RDF feeds from the following sites:

- Student Centre, University of Zagreb portal[4] – (news feed)
- Croatian music site[5] – (news feed and upcoming concerts feed)
- KSET: Electrical Engineering Students' Club[6] – (upcoming events feed)

Besides general entertainment information, portal offers a full personalized schedule of the student's study obligations (e.g. lecture schedule, exam dates, enrolment dates, etc…). Information about study obligations is gathered from several faculty information systems. Usage of various e-learning systems led to the fragmentation of information about student study. In order to retrieve all needed data related to one course, sometimes student has to use three or four different Web applications (one containing lecture schedule, the other containing midterm dates, etc...). As a basic personal service, users create an overview on all of their activities through importing dates of interest (entertainment events, exams dates, etc..) into their calendars (e.g. Google calendar). In addition, as the aspect of socialization, users can leave comments on news and participate in the forum.

[4] www.sczg.hr
[5] www.muzika.hr
[6] www.kset.org

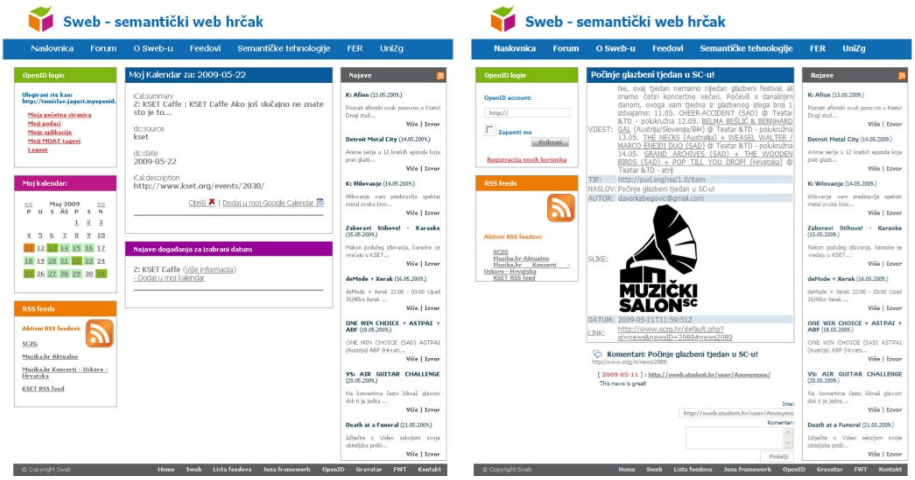

Figure 7: Sweb - Student semantic Web portal

The great difference in handling entertainment information and information about student study is in the required level of data accuracy and privacy. During the development of personal services, that require high level of data accuracy and security, semantic Web technologies did not prove to be up to a challenge.

5 Personalization and customization

Customization and personalization are two important components of a modern Web portal. Unfortunately, while customization is relatively easy to accomplish with different Web modules for each user group or role, personalization is much more difficult. Additionally, the problem we encountered was that very limited set of data and information was available for meaningful personalization. At the moment, relatively small percentage of overall Web data is stored in the RDF, so conversion from other formats ("on the fly" or "in advance") is necessary. Therefore, to supply our portal with the necessary content, we created RSS/RDF 1.0 feeds on a several existing Web sites, made GRDDL transformation algorithms and opened SPARQL endpoints which were then used as the RDF sources for data collection and manipulation. Unfortunately, due to the highly unstructured and inconsistent data in data models of the existing Web sites, we had to put an enormous effort in order to extract the useful data. As a result, most of the collected data does not provide rich set of information which can be used for profound or fine-grained personalization (eg. news article about a concert does not contain information about music genre or singer/band homepage). Existing tools and data collections (e.g. dbpedia[7]) are still not powerful and rich enough to easily supplement this gap, especially considering that we use Croatian language in our portal and most of the available data is in English.

[7] http://dbpedia.org/

Our implementation is based on the user's preferences, surfing habits and created data. Users can create a list of topics that can be used for fuzzy search or approximate analysis (e.g. if news articles and blog posts are described with MOAT [Passant, 08]). Also, the system tracks and counts various user actions and later on can suggest same or similar information. Analysis of the user's behaviour is a common problem [Schmidt, 07], thus we note the need for standardized "user behaviour" ontology. Furthermore, more complicated inferences can be made from collected forum data, comments (described using SIOC[8]) or meta data produced by user's friends (connected with foaf:knows attribute) assuming they share the same interests.

6 Data integration

Realization of the Semantic Web vision, caused by the technological inventions, is disjoined in two directions: "Web of tags" and "Web of data". Web 2.0 technologies, primarily Ajax (*Asynchronous JavaScript and XML*), made high user interactivity possible. Even in the non semantic applications, tagging, commenting, reviewing and similar user actions became very popular. In the "Web of tags", user is in the centre, it is she/he who creates data, gives it a well defined meaning and at the end, the one who consumes such data. The other direction is the one towards the "Linked data". Here, developers give meaning to data in the data layer, so wherever that data is presented, it will have the same meaning and every case of ambiguity is eliminated. Without a doubt, the latter path leads to a more complete and more precise solution. However, we believe that the global solution is in combining these two approaches so we have built Sweb in such a manner.

Sweb has two types of content: user created and application generated. To give a meaning to the application generated data we use technologies and methods from the "Linked data" approach. On the other hand, the entertainment information is gathered through RSS/RDF feeds, with data annotation performed during the feed creation. In that area we can not guarantee the information accuracy, but as we do not consider it sensitive, the integration can be achieved using similarity-based approaches. On the other hand, it is necessary that the information about student's study is accurate (eg. time-table, exam dates, etc.), in order not to cause any mishaps. Information about study has been gathered from the legacy databases and published according to "Linked data" principles, using the D2R for data conversion and setting a SPARQL endpoint. We also had a specific use case where it was much easier to provide GRDDL transformation algorithm than to introduce semantics in the data layer. When using GRDDL transformation the one works only with the explanatory semantic information, as that kind of information is usually the only presented to the user. The lack of identifying information tends to be a problem in the integration process. Although we have used the same vocabularies for all of the sources, the problem was in mapping the instances, especially when exact integration is an imperative. Our finding was that the easiest approach, and the one that promises scalability, was in the *partial centralization of the data*. Used approaches are explained in Section 6.2.

---

[8] http://sioc-project.org/

Integration process is carried out in the following steps: 1) retrieve data, 2) transform data, 3) merge data. Second step is needed only when the fetched data is not in the RDF, but in some other form.

6.1 Retrieving and transforming data

Portals usually cover a specific domain and, according to current Semantic Web standards, a set of data sources they use has to be explicitly defined. It is important to ensure that the application functions well even when the data sources are unavailable, which is the scenario that must be taken into account. Ensuring continuous functioning of the application in the case of the data source failure is one of the architectural requirements we have set - the reliability. As we previously discussed, the data sources that portal uses can be a protected access data sources. Semantic retrieval of the publicly available data is unambiguous; it is a SPROT query or a service (eg, GRDDL, RSS / RDF feed, etc.). However, access to protected data sources is different and requires a further explanation. For the protected data access we suggest the identity management system proposed in this research. In this case, the user is the one who initiates communication with an application that contains protected data, and all the other communication takes place between two applications, without user intervention. Figure 8 shows the communication between a semantic Web portal (Web application 1) and the protected application (Web application 2) using the proposed system for identity management.

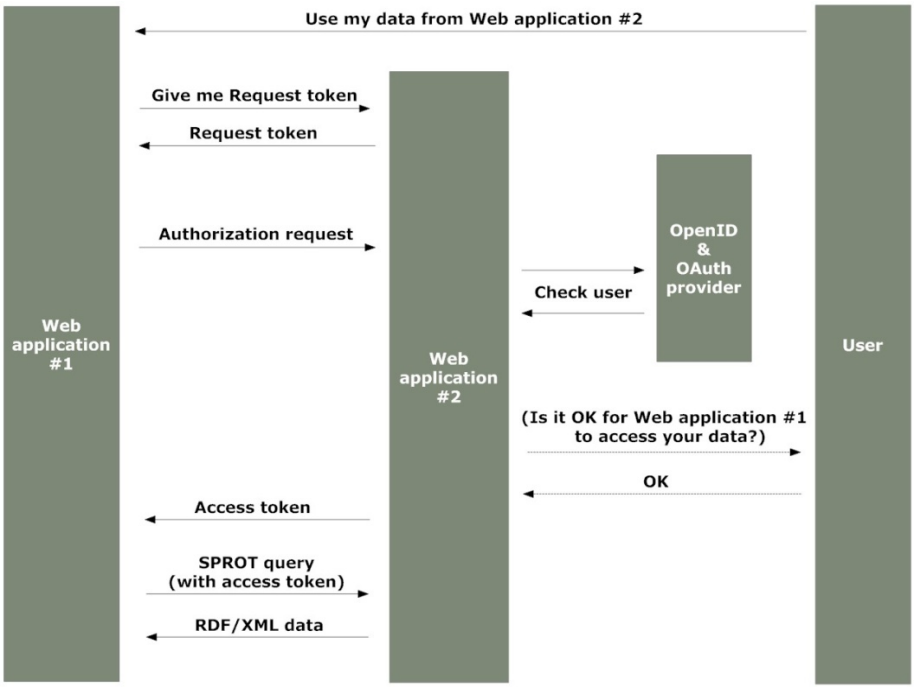

Figure 8: Web applications communication

6.2 Merging data

After the transformations were done, all of the data was available as a set of RDF graphs. The next step was to connect these graphs into a single RDF graph, which was then used in our application. Although, most of the resources used in Sweb are described using the vocabularies we provided, model discrepancy emerged and creating unified model proved to be a problem. Throughout development we used only a dozen of vocabularies, mostly standard ones (e.g. FOAF, SIOC, dc, iCal, etc.). Even on such a small set of vocabularies, and with good domain understanding by developers, bridging the data was a demanding task. Besides setting connections on the schema level, links had to be set on the instance level as well. Integration on the instance level proved to be a major problem for the data that needed complete accuracy as explained in the following text. Depending on the data type we propose the following integration mechanisms:

1. ***Personal data*** - Integration of user's personal data islands is easily achieved by introducing a *global unique identifier.* Used in the combination with single sign-on mechanism, it provides unambiguous integration process that keeps the data privacy and security uncompromised.
2. ***Data requiring the full accuracy*** – It is possible for two entities to have all property values equal, besides the identifying URIs, but still represent two different real world objects. For example, there are different courses with different IDs, but with the same name, number of ECTS, enrolment semester, lecturer, etc… Obviously, none of the common interlinking methods (similarity checks) solves this case. We could not find any appropriate solution that provides autonomy of the sources in the semantic data publishing process, and at the same time to be completely automatic, yet accurate. Our solution composes of imposing constraint on the data publishing process. All entities that could be ambiguous use just one *global identifying URI*. The problem is how to choose a global URI? In our case study, for every entity included in multiple applications it can be determined which application is an "entity owner", it is the application where such entity is being created. The URI published from that application is chosen as the *global identifying URI.* The result of this approach is scalability, as no need for manual mapping is required when a new source is added.
3. ***Data not requiring the full accuracy –*** When ambiguity in the data integration is acceptable, any kind of similarity check on the entity properties can be implemented. So, to allow user to participate in the data mapping process, we introduced annotation (tagging) of the news and the forum content. It was implemented with MOAT. The links made in such manner, of course, could not be taken with certainty as they rely on user's accuracy. However, this approach had additional value, as stated in the [Hausenblas, 08], allowing users to generate content can speed up the semantic enrichment of the Web.

7 Architecture overview

While creating Sweb, we verified the proposed architecture of a Semantic Web portal. Since Sweb integrates publicly available information and the protected user data, through its production, the identity management system and proposed content exchange protocols were also verified. The implementation showed to what extent the application architecture can be set to meet technical requirements at the moment. Overview of the trade-offs we had to make in the architecture are showed in the Table 7. Table shows that the security requirement is satisfied. The main problem is lack of access control on the permanent data stores, as well as access to the SPARQL endpoint. The proposed data exchange mechanisms partially solves the problem, but it also would have significant benefit from access control mechanisms. Maintenance and scalability are affected by the number of non-semantic data sources. For each data source we had to write a separate transformer which converts data into the RDF.

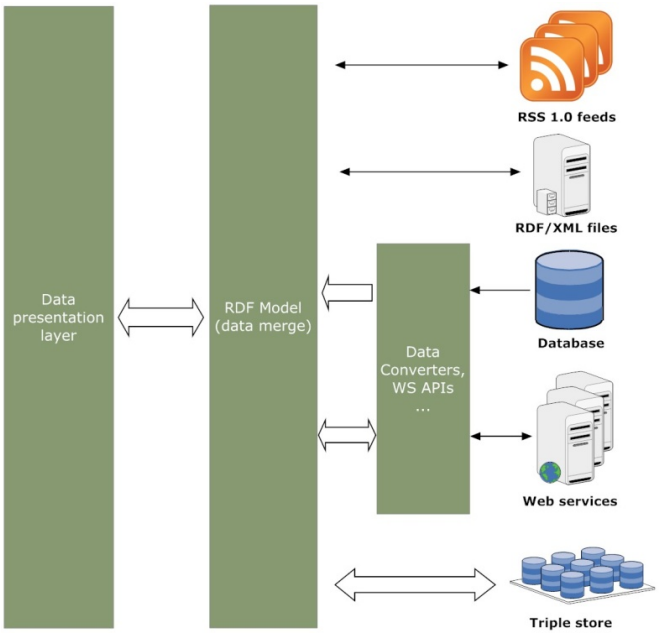

Figure 9: Sweb high-level architecture

Data integration has a considerable impact on maintenance and scalability. Since it is impossible to use the automatic integration, because the full data accuracy is required, each change of the data source requires manual update of the ontology mapping, and, in some cases, an agreement and Unification of different identifiers. To implement the content tagging feature in the Sweb, the MOAT server was created. Communication between the MOAT server and Sweb was accomplished as in the Figure 5, and its deployment did not reveal any possible problem. The implemented identity management system has a potential single point of failure. In the event of system

failure users will be denied access to multiple applications. We paid special attention on providing usability, as our opinion is that current Semantic Web technologies are not suited for general public, and usually require some basic knowledge of Semantic Web purposes. There is no any visible sign of semantic technologies on the application interface, although portal is completely based on it. Presented data is annotated (wherever possible) using RDFa (Resource Description Framework attributes) [Adida, 09], so any RDFa aware browser or browser plug-in (e.g. Semantic Radar) can make use of it. Using RDFa in presentation layer enables interoperability. To additionally support interoperability we put an effort in opening access to our user generated content for other services, as we already prepared it for integration using SIOC and MOAT. But, we could not find any other service to communicate with. There is also export feature for user's data (in RDF/XML or vCard format), and a number of available RSS 1.0 feeds, which could be used in a similar way we used RSS feeds in our project.

	maintainability	**Security**	**usability**	**Scalability**	**robustness and reliability**	**Interoperability**
single sign-on	not part of W3C standards; possible compatibility problems	no trade-offs	no trade-offs	no trade-offs	single point of failure	no trade-offs
data storage	no trade-offs	lack of control access solution	need for better administration tools	not tested on large triple number	not tested	no trade-offs
RSS/RDF, GRDDL data aggregation	hard to achieve generalization, usually needs custom handling	only for publicly available data	information could be delayed due to caching	new source needs custom touch	assured handling of possible source failure	no trade-offs
SPPARQL endpoint data aggregation	no trade-offs	not part of W3C standardization	no trade-offs	no trade-offs	possible problems on large databases using D2R	no trade-offs
Data integration	need for reconciliation on higher level (partial data centralization)	not part of W3C standardization	not fully automated	need for additional custom mapping	possible failure in case of source structure changes	no trade-offs
Data tagging	no trade-offs	no trade-offs	no trade-offs	no trade-offs	no trade-offs	no trade-offs
User interface	no trade-offs	no trade-offs	no trade-offs	no trade-offs	no trade-offs	no trade-offs

Table 2: Main architectural aspects

8 Related work

Even though a significant work has been done in the semantic Web portals development area, since the Semantic Web vision has changed and semantic technologies followed that change, none of the semantic Web portal solutions can be taken as a relevant and serve as a prototype. For instance, in the survey on the

semantic Web portals that was taken in the year 2004 [Lausen, 05], none of the solutions under consideration deals with the heterogonous data sources integration. The MuseumFinland project [Hyvönen, 05] is a typical implementation of the semantic portal pattern, but still does not support dynamic integration and data retrieval. [Bellekens, 07] presented an approach that resembles ours in a dynamic data integration and personalization, but does not tackle the integration of the personal data.

The work of [Hausenblas, 08] has the most resemblances to our own work. They emphasize the important issues regarding the synergy of the semantic Web and Web 2.0 and argue the usability of non reliable data interlinking. However, as they consider only publicly available data, we could not base our approach on their findings.

The importance of identity management and single sign-on in the Semantic Web context has already been recognized and intensive research in this field is in progress. [Bojars, 08; Mostarda, 09] The idea of using FOAF for describing user profile was already presented in several studies [Ankolekar, 06; Grzonkowski, 05; Kruk, 04]. Our approach is not limited on using just one vocabulary as it is very likely that FOAF will not be enough when we exceed this proof-of-concept phase.

We argue that e-mail address cannot be used as an identifier, which is suggested in approaches [Grzonkowski, 05; Kruk, 04] as it is sometimes considered to be a private data. Further, as we prioritize privacy and security as well as correctness of the data, we fully entrust user in setting the explicit access rules. Approach suggested in [Grzonkowski, 05; Kruk, 04] can be used in social applications not containing any sensitive data since calculating the trust between people based on “percentage of friendship” is error prone and not controlled by the user. In our approach users define explicit access rules for the applications consuming user profile data.

9 Conclusion

The main goal of this research was to develop a solution that would automate user’s repetitive tasks while using the Web. Our research has shown the basic functionalities that have to be implemented in order to achieve such solution. These are: data aggregation, data integration and personalization. We have recognized that the Semantic Web technologies could facilitate the implementation of these functionalities, so we have proposed semantically enhanced Web portal. As these technologies are still rather new, we set architectural requirements that our solution had to satisfy in order to evaluate maturity and completeness of the Semantic Web technologies as its building blocks. We argue that the major drawback at the moment is security, more precisely, assuring data privacy and protection. To achieve an automated authenticated access to user’s private data, we introduced an identity management system for the Semantic Web environment and proposed several extensions of the existing technologies. We have recognized OAuth, alongside well standardized Semantic Web technologies, with the SPARQL protocol as the fundamental technology for our approach. Our solution was verified by creating a prototype of a semantically enhanced Web portal that uses developed identity management system to achieve authorized access to user’s personal data.

Future work encompasses integration of more advanced access control framework instead of the one we developed only to prove our hypothesis. We also intend to enlarge the number of data sources portal relies on, as the lack of data significantly restrained us in estimating personalization process quality.

Acknowledgements

The presented work was funded by Croatian Ministry of Science, Education and Sports, record no. 036-0361983-2012. The authors would like to thank the funding agency for their support.

References

[Adida, 09] Adida, B. and Birbeck, M., RDFa primer 1.0: Embedding rdf in XHTML, 2009, www.w3.org/TR/xhtml-rdfa-primer/

[Ankolekar, 06] Ankolekar, A. and Vrandecic, D., "Personalizing Web surfing with semantically enriched personal profiles", Proc. Semantic Web Personalization Workshop, (Budva, Montenegro, 2006).

[Atwood, 09] Atwood, M., Conlan, R., et. Al., OAuth Core 1.0, 2009, http://oauth.net/core/1.0/

[Battle, 08] Battle, R. and Benson, E. "Bridging the semantic Web and Web 2.0 with Representational State Transfer (REST)", Web Semantics: Science, Services and Agents on the World Wide Web, 6 (1) (2008), 61-69.

[Bellekens, 07] Bellekens, P., Aroyo, L., Houben, G., Kaptein, A. and Van Der Sluijs, K. "Semantics-Based Framework for Personalized Access to TV Content: The iFanzy Use Case", Lecture notes in computer science, 4825(2007), 156 - 165.

[Berners-Lee, 10] Berners-Lee, T.: 'Linked data', http://www.w3.org/DesignIssues/LinkedData.html, 2.9.2010.

[Bizer, 09a]Bizer, C., Cyganiak, R., Heath, T.: 'How to publish linked data on the Web', http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/

[Bizer, 09b] Bizer, C. and Cyganiak, R. "Quality-driven information filtering using the WIQA policy framework", Web Semantics: Science, Services and Agents on the World Wide Web, 7 (1) (2009), 1-10.

[Bizer, 09c]Bizer, C., Schultz, A.: „The berlin sparql benchmark“, International Journal On Semantic Web and Information Systems-Special Issue on Scalability and Performance of Semantic Web Systems, (2009)

[Bojars, 08] Bojars, U., Passant, A., Breslin, J. and Decker, S., "Social network and data portability using semantic Web technologies", Proc. BIS (Workshops), (2008), 5-19.

[Brickley, 09] Brickley, D. and Miller, L., FOAF vocabulary specification 0.91, 2009, http://xmlns.com/foaf/spec/

[Clark, 09] Clark, K., SPARQL protocol for RDF, 2005, www.w3.org/TR/rdf-sparql-protocol/

[Clarke, 04] Clarke, R. Identity Management: The Technologies, Their Business Value, Their Problems, Their Prospects. Xamax Consultancy Pty Ltd, Melbourne, Australia, (2004).

[Dietzold, 06] Dietzold, S. and Auer, S., "Access control on RDF triple stores from a semantic wiki perspective", Proc. 2nd Workshop on Scripting for the Semantic Web at ESWC2006, (Budva, Montenegro, 2006).

[Fitzpatrick, 09] Fitzpatrick, B. et al., OpenID Authentication 2.0, 2009, http://openid.net/specs/openid-authentication-2_0.html

[Grzonkowski, 05] Grzonkowski, S., Gzella, A., Krawczyk, H., Kruk, S., Moyano, F. and Woroniecki, T., "D-FOAF-Security Aspects in Distributed User Managment System", Proc. IEEE International Conference on Technologies for Homeland Security and Safety (TEHOSS 2005), (Gdansk, Poland, 2005).

[Hassanzadeh, 09] Hassanzadeh, O. and Consens, M., "Linked Movie Data Base", Proc. 2nd Linked Data on the Web Workshop (LDOW2009), (2009).

[Hausenblas, 08] Hausenblas, M., Halb, W. and Raimond, Y., "Scripting User Contributed Interlinking", Proc. 4th Workshop on Scripting for the Semantic Web (SFSW 08), (Tenerife, Spain, 2008).

[Hyvönen, 05] Hyvönen, E., Mäkelä, E., Salminen, M., Valo, A., Viljanen, K., Saarela, S., Junnila, M. and Kettula, S. "MuseumFinland—Finnish museums on the semantic Web", Web Semantics: Science, Services and Agents on the World Wide Web, 3 (2-3) (2005), 224-241.

[Jain, 06] Jain, A. and Farkas, C., "Secure resource description framework: an access control model", Proc. Eleventh ACM symposium on Access control models and technologies, (2006), ACM New York, NY, USA, 121-129.

[Kruk, 04] Kruk, S., "Foaf-realm-control your friends' access to the resource", Proc. 1st Workshop on Friend of a Friend, Social Networking and the (Semantic) Web (FOAF Galway), (Galway, Ireland, 2004), 1-9.

[Lausen, 05] Lausen, H., Ding, Y., Stollberg, M., Fensel, D., Hernández, R. and Han, S. "Semantic Web portals: state-of-the-art survey". Journal of Knowledge Management, 9 (5) (2005). 40-49.

[Liang, 10] Liang T-P, Yang Y-F, Chen D-N, Ku Y-C. A Semantic Expansion Approach to Personalized Knowledge Recommendation. Decision Support Systems 2008; 45(3): 401-412

[Manjunath, 09] Manjunath, G., Sayers, C., Reynolds, D., KS, V., Mohalik, S., Badrinath, R., Recker, J. and Mesarina, M., Semantic Views for Controlled Access to the Semantic Web, 2009, http://www.hpl.hp.com/techreports/2008/HPL-2008-15.html?mtxs=rss-hpl-tr

[Miyata, 06]Miyata, T., Koga, Y., Madsen, P., Adachi, S., Tsuchiya, Y., Sakamoto, Y., Takahashi, K.: 'A survey on identity management protocols and standards', IEICE TRANSACTIONS on Information and Systems, 2006, 89, (1), pp. 112-123

[Mostarda, 09] Mostarda, M., Palmisano, D., Zani, F. and Tripodi, S., "Towards an OpenID-based solution to the Social Network Interoperability problem", Proc. W3C Workshop on the Future of Social Networking, (2009).

[Murugesan, 07] Murugesan S.: "Web Application Development: Challenges and the Role of Web Engineering". in "Web engineering: modelling and implementing Web applications", Springer, (2007), 7-32.

[Notoatmodjo, 07] Notoatmodjo, G. Exploring the 'Weakest Link': A Study of Personal Password Security. , MSc Thesis, Computer Science Department, The University of Auckland, New Zealand, (2007).

[OpenID, 09] OpenID, 2009, http://openid.net/

[Passant, 08] Passant, A. and Laublet, P., "Meaning Of A Tag: A collaborative approach to bridge the gap between tagging and Linked Data", Proc. Linked Data on the Web, (2008).

[Perry, 92] Perry, D. and Wolf, A. "Foundations for the study of software architecture", ACM SIGSOFT Software Engineering Notes, 17 (4) (1992), 40-52.

[Raimond, 08] Raimond, Y., Sutton, C. and Sandler, M., "Automatic interlinking of music datasets on the semantic Web", Proc. Linked Data on the Web Workshop (LDOW2008), (2008).

[Reddivari, 07] Reddivari, P., Finin, T. and Joshi, A., "Policy based access control for a rdf store", Proc. IJCAI-07 Workshop on Semantic Web for Collaborative Knowledge Acquisition, (2007), 78–83.

[Samar, 99] Samar V., "Single sign-on using cookies for Web applications". Proc. IEEE 8th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, (WET ICE'99), (1999), 158-163.

[Schmidt, 07] Schmidt, K., Stojanovic, L., Stojanovic, N. and Thomas, S., "On enriching ajax with semantics: The Web personalization use case", Proc. 4th European conference on The Semantic Web: Research and Applications, (Innsbruck, Austria, 2007), 686-700.

[Suriadi, 09] Suriadi S., Foo E., Jøsang A.: "A user-centric federated single sign-on" system. "Journal of Network and Computer Applications", 32 (2) (2009), 388-401.

[Volz, 09] Volz, J., Bizer, C., Gaedke, M. and Kobilarov, G., "Silk–A Link Discovery Framework for the Web of Data", Proc. 2nd Linked Data on the Web Workshop (LDOW2009), (2009).

[Vuljanić, 10] Vuljanić, D.; Rovan, L.; Baranović, M., 'Semantically Enhanced Web Personalization Approaches and Techniques', In: Lužar-Stiffler, Vesna; Jarec, Iva; Bekić, Zoran, editors. Proceedings of 32nd International Conference on Information Technology Interfaces, 2010 Jun 21-24, Cavtat, Croatia. Zagreb: SRCE University Computing Centre, University of Zagreb; 2010. pp. 217-222

[Wang, 06]Wang, C., Lu, J., Zhang, G.: 'Integration of ontology data through learning instance matching'. In: Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence 2006, pp. 536-539