Gnowsis Semantic Desktop ISWC2004 Demo

Leo Sauermann
DFKI GmbH Knowledge Management Group
Kaiserslautern, Germany
leo.sauermann@dfki.de

Abstract

We present our vision of a Semantic Web enhanced desktop environment, the *Semantic Desktop*. Existing desktop resources like MS-Outlook contacts or Mozilla bookmarks are treated as Semantic Web Resources. The *Gnowsis* system is our open source project that implements this vision. It is released in September 2004 as alpha version – to be used and extended by Semantic Web developers.

1. Introduction

The Semantic Web has not lifted off the ground yet. Tim Berners-Lee tells us again and again to start writing useful applications [Berners-Lee 2003]. While many projects focus on Semantic Web services and server based systems with focus on the "big picture", our Gnowsis Semantic Desktop approach is targeted to organize files, emails and other items on personal computers. At the moment the most successful (in terms of publicity) projects are build on small but effective schemas like *Friend-Of-A-Friend* (FOAF)[1] or *RDF Site Syndication* (RSS)[2]. We also want to take small steps. The use of Semantic Web technology in desktop applications will create a benefit for users. When users benefit from it, the technology will be used and resources will be annotated. When these annotated resources are published, the Semantic Web will lift off. It is also a good learning playground for researchers – the experience we gather from organizing our own files and emails using the *Resource Description Framework* (RDF) will be useful for building a global Semantic Web.

Personal computers are gaining an important role in our daily life. They are used to view images or DVDs, to listen to music, to communicate with others via email or instant messaging, and to organise our everyday life via calendars and address books. Most of these tasks are done by standalone applications. One application does not know about the data stored in other applications or other files it cannot read. On the other hand, all the resources stored on a desktop computer have semantic relations, a person stored in an address book may attend an appointment in the calendar or be depicted on a photo. A friend can give an MP3 song to you, so the song is related to the person.

All these information items can be treated as Semantic Web resources. An email is also a Semantic Web resource. Its data can be represented as RDF resource in an RDF graph. Existing technology and approaches from the Semantic Web can be used on these desktop resources. The semantic relations between the resources can be represented as RDF triples. When this representation of information as RDF graph exists, a Semantic Browser can be used to visualize the information and allow the user to see how his or her information is related. This results in a Semantic Web enhanced Desktop – the *Semantic Desktop*. Applications still handle only their part of the whole – and integrate into a bigger view through RDF.

A look at current business practice showed us that companies want to integrate their applications, a common task known as *Enterprise Application Integration* (EAI). In combination with *Knowledge Management* ideas (KM), the result is often an intranet portal website that employees use to access their different information sources. We chose a similar approach to build the *Gnowsis* system. Taking existing information from existing applications and integrating it by using a relatively small amount of *glue code*.

A Semantic Web application is useful when it builds on real life data that the users already have. So many developers are implementing "*glue code*" adapters to access data as RDF graph or convert data into RDF. These adapters have to be reused. The Gnowsis system includes a framework for taking data from arbitrary data sources and representing them as RDF triples. Developers can use existing Gnowsis adapters or can integrate their own adapters into the Gnowsis framework. We don't want to invest time on converting data anymore, we want to build applications.

---

[1] http://www.foaf-project.org
[2] http://purl.org/rss/1.0/

2. Implementation

The popular RDF framework Jena[3] by HP Labs was chosen as the main technical ground to build Gnowsis on. We extended the Jena framework with the Semantic Desktop functions. For different data sources like the file-system or Microsoft Outlook we created adapters, which are accessible as Jena models. We also integrated existing adapters, for relational databases we used *D2RQ* from Chris Bizer [Bizer 2004]. Developers and Researchers can use these Jena Models in their own applications to extract information from the different applications.

2.1 Data Integration - CentralHub

The basis of the architecture is the data integration framework. We wanted to access all data of the desktop computer through a single RDF graph. This graph is called the "*CentralHub*" and is consists of several sub-graphs, the *Adapters*. For each data source, a seperate Adapter was built. For example, to access the Filesystem as RDF graph, there is a FilesystemAdapter. Each adapter is a Java class that implements the Jena Graph Model. So each adapter can also be used in your Java applications like a Jena Model. Each adapter is shipped with a RDF-Schema (RDFS) vocabulary that describes what data is available in the Adapter. The adapters read data from the native data sources only when needed and do the conversion to RDF on the fly. All adapters are implemented read-only at the moment. We used common RDFS vocabularies to represent the adapted data. For example, both Outlook and Mozilla addressbook entries are represented as FOAF-Persons.

All adapters are integrated in the *CentralHub*, which itself is also a Jena Model. It answers queries by forwarding them to the different adapters. The adapters than extract the triples from the native data sources and return them. To decide which adapter knows information about a given resource, the *URIs* of the resources are parsed. Like an Apache web-server decides to pass a request to the PHP module by evaluating the request URL for the ".php" file extension, we mapped adapters to URI schemes. This approach is very simple but we learned that exactly this simplicity made the WWW such a success. The *CentralHub* also parsed RDFS information from the request, to answer queries like "`* <rdf:type> <foaf:Person>`".

To build such a system, an experimental approach in the Semantic Web field had to be taken. While people still discuss the meaning and interpretation of URIs on the RDF-IG newsgroup and the W3C [Berners-Lee 2002], we tested different. All resources on the desktop are identified by a URL, be it a file, an email or a database row. These URLs can be used to query information about the Resources or to annotate the resources using new RDF triples. (We do not use the "Inverse

Functional Property" approach that is social practice in many FOAF files).

The CentralHub system is generally **read-only**. We can access the information from applications like Outlook or a relational database, but we cannot change it. To change this information, we recommend to open these applications to edit the resources, this is supported in our user interface "*Enquire2*" (see below).

But we can **store additional triples** in a central RDF database. Using it, it is possible to store additional triples that link existing resources or to create new resources. Developers can use the central RDF database to store custom data. We also implemented a tagging API here that allows us to mark subgraphs in the database. Using the tagging API it is easy to update resources by deleting a subgraph and adding a new version of it. The whole database is again a Jena Model that is backed by an RDF/XML file. Storage can be exchanged with a database, if needed.

2.2 API to the Gnowsis

There are many ways to exchange RDF data between applications or how to query them from servers. Available implementations are the Joseki Server[4] by HP Labs or Sesame[5] which both provide HTTP interfaces. We decided to use RDQL[6] to find resources and an extension of Concise Bounded Descriptions (CBD) [Stickler 2004] to retrieve the RDF data.

The Gnowsis system includes a Gnowsis Server that is installed on the Semantic Desktop computer. Developers can access the CentralHub functionality through XML/RPC interfaces using a *client Application Programmer Interface* (API). The client API is again based on Jena.

2.3 GUI Integration

We focused on writing a simple RDF browser that we called "*Enquire2*". This browser can load the RDF description of a resource and display it. It also displays related resources. It is possible to link resources to each other just by drag-dropping one resource over another. The links are created using predicates taken from RDFS vocabularies. See Figure 1 for a view of a person resource (as an example the author was taken).

To embed the Gnowsis functionality into existing graphical user interfaces of existing applications, we took a very effective approach. Basically, it is much work to implement extended functionality into 3rd Party applications like MS-Outlook, Mozilla, Konqueror, the Windows Explorer or OpenOffice.

We want to enable people to use their existing applications in combination with the RDF features. To integrate Gnowsis, we just added two buttons to existing applications: **link and browse**. The browse button opens the selected resource in the Enquire2 browser

---

[3] http://jena.sourceforge.net/

[4] http://www.joseki.org/

[5] http://www.openrdf.org/

[6] http://www.hpl.hp.com/semweb/rdql.htm

and shows related resources. The link button tells the system to link this resource to another resource. To link two arbitrary resources, the user presses "link" at both resources, in their native applications.
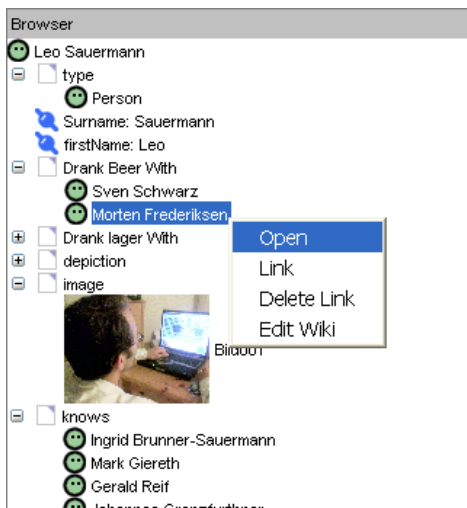

Figure 1 Browsing a person

Figure 1 shows the resources that are related to a person. As an example, the person Leo Sauermann is related to Morten Frederiksen by the fact of mutual beer drinking. Note the popup-menu that allows the user to open the MS-Outlook address-card of Morten Frederiksen.

2.4 Typical Use Case

A typical use case is that the user wants to annotate that an image depicts a person. Right-clicking the image file allows the user to "link" it in his Semantic Desktop. The request is sent to the local Gnowsis server and the server starts Enquire2 to show the image.

Then the person depicted on the image can be selected in Outlook using the fast search capabilities of Outlook. The user presses "link" at the address card of the person and again this command is passed to Gnowsis. In Enquire2, both resources are shown and the user can select how the resources are related.

After the relation is entered to the system, it can be displayed by clicking the "browse" button on either the image or the person.

Files can also be drag-dropped on the Enquire2 browser, so drag-dropping an image on a person creates the same relation.

3. Demo Presentation

At the ISWC 2004 we plan to present an installed version of the Gnowsis system. We will explain the system based on a real life example: How it is possible to organize the trip to the ISWC and annotate people, presentations, papers and other related material to the conference.

Attendees of the presentation can download the Gnowsis system from the website and install it on their own desktop systems. See below for the status of the project.

Figure 2 Gnowsis buttons in MS-Outlook

4. Related Work

There are different approaches with similar goals. Haystack [Quan *et al.,* 2003] is similar to Gnowsis. Haystack is also a framework to build RDF applications, having its own user interface and RDF engine.

Another related project is the EPOS project by the DFKI. It aims at evolving personal information models to organizational knowledge spaces. A part of EPOS is eliciting the user's current context in order to relate his current work to his desktop resources. It also identifies current working steps (workflow tasks) [Schwarz et al, 2003].

Gnowsis is different from the related work because we focus on integrating existing applications instead of replacing them. This approach is also followed in the DFKI EPOS project. In today's software projects it is a common demand that new software integrates with Microsoft's Outlook or popular tools like the Mozilla suite.

5. About the project

The Gnowsis system was first created in Vienna by Leo Sauermann as his diploma thesis. Since July 2004, he is member of Knowledge Management department of the German Research Center for Artificial Intelligence (DFKI) and Gnowsis is hosted there. The DFKI uses Gnowsis in several research projects.

The project is hosted and documented on the website http://www.gnowsis.org. In September 2004 we release an alpha version. The source code and Java binaries will be available under an BSD compatible open source license.

The framework is Java based and currently specially adapted to **Windows**. We are working on **Linux** and **MacOs X** integration. Most of the system already runs under Linux and Mac-OS, only the drag-drop features and some other GUI related problems have to be solved.

We will continue the development and plan to create better user interfaces. We also plan to implement col-

laboration features in order to create a Networked Semantic Desktop.

The Next Steps

We plan to enhance the integration of the Gnowsis into existing systems. The adapters should not be read-only, we want to also allow write access to the data sources. But this needs some basic research. E.g. we do not know how to represent a "move File X" operation just by storing RDF triples, an RDF-DIFF [Berners-Lee et al, 2004] approach is needed.

The Semantic Desktop will play a key role in building a global Semantic Web. First users will annotate their personal data, motivated by the benefits of an integrated personal information management. Then this annotated data can be published on the web.

Web sites don't have to be annotated with the "add semantic sugar on top" philosophy, instead all resources can be annotated starting with the time of creation.

Once a file on the desktop is annotated, it will be a small step to include the annotations when the file is sent through email, posted on a website or stored in an organizational memory. Then other users can access the semantic annotations and benefit from them.

References

[Berners-Lee et al, 2004] Tim Berners-Lee, Dan Connolly, *Delta: an ontology for the distribution of differences between RDF graphs*
http://www.w3.org/DesignIssues/Diff
last change: 2004/05/01

[Berners-Lee 2003] Tim Berners-Lee, *Where to Direct Our Energy* ,2nd International Semantic Web Conference, October 23, 2003
http://www.w3.org/2003/Talks/1023-iswc-tbl/Overview.html

[Berners-Lee 2002] Tim Berners-Lee, "What do HTTP URIs identify?", July 2003
http://www.w3.org/DesignIssues/HTTP-URI

[Bizer 2004] Chris Bizer, *D2RQ - Treating Non-RDF Databases as Virtual RDF Graphs*
http://www.wiwiss.fu-berlin.de/suhl/bizer/d2rq/index.htm

[Quan *et al.,* 2003] Dennis Quan, David Huynh, and David R. Karger. *Haystack: A Platform for Authoring End User Semantic Web Applications*. International Semantic Web Conference 2003: 738-753

[Schwarz et al, 2003] Sven Schwarz and Thomas Roth-Berghofer, *Towards Goal Elicitation by User Observation*, Proceedings of the FGWM 2003 Workshop on Knowledge and Experience Management

[Stickler 2004] Patrick Stickler. *URIQA the Uri Query Agent Model*. Contains also the methodology of Concise Bounded Descriptions (CBD)
http://sw.nokia.com/uriqa/URIQA.html

For more references, see the URLs in the footnotes.