# ‘SemanticLIFE’ – A FRAMEWORK FOR MANAGING INFORMATION OF A HUMAN LIFETIME

Ahmed M., Hoang H.H., Karim M.S., Khusro S., Lanzenberger M., Latif K., Michlmayr E., Mustofa K., Nguyen H.T, Rauber A., Schatten A., Tho M. N., Tjoa A M.
Vienna University of Technology
Favoritenstrasse 9-11/E188
A-1040 Vienna, Austria
+43 1 58801 18801
schatten@ifs.tuwien.ac.at

*Abstract*

*The ‘SemanticLIFE’ system is designed to store, manage and retrieve ones lifetime’s information entities. It enables the acquisition and storage of data while giving annotations to email messages, browsed web pages, phone calls, images, contacts, life events and other resources. ‘SemanticLIFE’ also provides intuitive and effective search mechanism based upon stored semantics. This paper presents the framework for the system to devise lifetime data store and search using recognized standards. We are aiming at a system which supports the long term memory by associating metadata with content and ontologies. The ultimate goal of our project is to build a Personal Information Management system over a Human Lifetime using ontologies as a basis for the representation of its content.*

## 1. Introduction

*‘A memex is a device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to his memory’ - Vanevar Bush*

*‘A journey of a thousand miles begins with a single step’* – Chinese proverb

The ‘SemanticLIFE’ project, which is introduced in this paper, is an attempt to come a step closer to Vanevar Bush’s vision of the Memex from the year 1945. Recently we can observe a mushrooming of new projects aiming at some of the goals of Bush’s innovative ideas. This is mainly caused by the racy technological development which opens new large realization potentials. An indicator for the narrowing of the discrepancy between the vision of Bush versus its realization is the announcement of ‘Memories for life’ -- Managing information over a human lifetime as one of the seven Grand Challenges for Computing Research by the UK Computing Research Committee. The announcement of the First ACM Workshop on Continuous Archival and Retrieval of Personal Experiences in October 2004 is another indicator for the maturity of our present time to implement such systems.

This research has partly been funded by the Austrian Federal Ministry for Education, Science, and Culture, and the European Social Fund (ESF) under grant 31.963/46-VII/9/2002.

Another significant development which narrows the gap toward the realization of memex-like systems is the advent of the Semantic Web initiatives. As proposed by Tim Berners-Lee [2] the realization of Semantic Web would narrow this gap by the use of domain specific ontologies and their reuse. The semantic web is in its early stages of development [8]. Although the most core issue of developing machine-processable meta-information [17] has been answered considerably by the emergence of RDF(S) [1], OWL [5] and Topic Maps [3], but the application aspects of semantic web still have to be explored.

The goal of our project is to build a Personal Information Management (PIM) system over a Human Lifetime using ontologies for the representation of semantics. Basic ontological infrastructure required for applications development on top of the current generation of the Web are very well described in [8].

This paper describes the architecture of the 'SemanticLIFE' project which aims at realizing a digital personal diary that records everything a person wants to be kept. This notion defines also the boundaries of our project – we do not deal with memory issues of the unconscious or procedural memories (e.g. how to open a bottle). We are aiming at a tool which supports the long term memory by associating metadata with content and ontologies. The possibility of adding annotations to all stored objects should enrich the potential use of such a 'diary'. The project as a whole covers aspects of Human Computer Interfaces, Databases, Data Mining, Security, Device Engineering etc. A special focus will also be laid on dedicated interfaces for people with special needs.

The project is invested as an Open Source enterprise using as much as possible recognized standards.

The first prototypes are intended to show the increase of suitability of searching by the use of ontologies for the managing information of human life time within the boundaries of the project described above. These prototypes are also restricted in this first stage for single user practice. Therefore aspects of privacy and authentication are not covered in this paper. Ethical issues are also beyond the scope of this paper – although will be part of our research work.

The very next section is about related work. In section 3 we describe the vision of 'SemanticLIFE'. Section 4 concludes the requirements for the system and proposes the design and architecture to achieve our goal. We are concluding our work in section 5.

## 2. Related Work

In the domain of semantic-web, numerous projects and tools are available, although most projects still are not proven for production use. Basically, the efforts fall into two major categories: *Ontology Management* and *Personal Information Management (PIM).*

A plethora of tools for ontology development and other related tasks, like ontology development, annotation, evaluation, merging, storage and retrieval, are available [4]. Some of these tools like *Protégé, Sesame, Jena, Ontomat* and *KAON* are possible candidates for use in our project for their respective application.

As far as PIM systems are concerned, a few tools based upon the semantic web technology are available. To give some examples: *MyLifeBits* is an ontological store where information entities enriched with annotations, documents, images and sound [11] can be stored. Lifelog is part of

DARPA’s research in cognitive computing, designed to extend the model of a personal digital assistant to one that might eventually become a personal digital partner. *Haystack,* by MIT is more of an ontological visualization solution [14]. HP Labs provides a framework called *e-Person,* a personal information infrastructure that enables groups of users to organize and share information [15][20]. Some other tools like *LifeStreams* [9], *Edutella* [19] and *Semagix Freedom* [21] cover certain aspects of an individual's life but not all of it or not in a complete semantic manner. In these systems a limited support for features like automatic feeding from different data sources, metadata extraction from information chunks, manual annotation and more importantly processing of Life Events is noticed.

## 3. ‘SemanticLIFE’

Living systems have different characteristics like *self-regulation of processes*, *reproduction* and *growth* [18]. Nevertheless, the relevant characteristics could be envisioned in a semantic way in personal knowledge management. Ontologies of personal life items grow and reproduce new ones with processes and services. These ontologies include information about our life objects such as documents, persons, places, organizations, events and tasks.

In the physical world, entities are usually interconnected, either by physical or by semantic means; in the latter case, the semantic meaning is added by human interaction (in an abstract sense) with the physical world. *Life items* in the system proposed in this paper can be understood as information entities (in some cases they are representations of such physical entities) stored according to ontologies in a semantic database, which are connected to other information entities according to their semantic meaning. Also ontologies ‘live’ in a way, as they develop and modify permanently during the system- and user- lifetime.

Current (Web-) technologies are highly efficient in processing data for human reception; that is, the transformation from data to information, the ‘generation of meaning’ is up to the human. A great deal of effort has already been made, and work is still going on to represent semantics explicitly on the Web. This is required to give computer systems the capability to enhance preprocessing of huge amounts of data for the user. It becomes more important as the ‘awareness radius’ of the contemporary knowledge worker and consumer is continuously increasing. This results from the observation, that users do not limit their information search to specific data repositories, like searching for an address or an event in a calendar any longer. The availability of databases under common or similar interfaces (like web-pages) creates the demand to express more complex queries demanding information aggregated from many different systems using different semantic concepts.

The proposed PIM systems can contribute significantly in overcoming the common inherent human problems such as limited short term memory, memory loss, forgetfulness, high complexity of data etc. Therefore, it is useful for the system to be able to define and capture the user’s life-related events and takes or triggers appropriate action(s) for it. This process involves the following sub-processes:

- Capture events and associated information
- Process action associated with events (e.g., in the sense of an active database system)
- Extract Metadata from the event, or allow the user to enrich the data manually with semantic meaning

- Store the data including semantic context as ontology in an efficient manner
- Allow the user to query the data or support the user directly or via associated applications and tools with context-sensitive information or action

The typical usage of such a PIM can be illustrated with two examples: A student searches a book written by a specific person knowing only a part of the title and the fact, that the author is a graduate of a specific university. The desired result is, e.g., a link to the book provided by an online bookstore, where the student is customer.

A second example: Consider scientists, who work in a specific domain. They might be interested to get into contact with other researchers in the scientific community that (1) share the same interests or have similar problems (2) are publishing in similar conferences and (3) were recently active in the specific field of research (4) and speak a common language. The result of such a query could be the web pages and email addresses of the researchers coming into question.

It is clear that such problems can only be solved by querying a multitude of information resources like web pages, conference journals, scientific databases, email repositories, newsgroups and the like. Moreover, the system needs to 'understand' that entities differently labeled are identical in a semantic sense and also need to be able to 'understand' and solve specific issues like the fact, that some results are only valid in a specific interval of time or in a specific language and so on.

Additionally as described in Dolog et.al. [7], the system must be able to adjust to new user features derived from user interactions with the system or from the information being fed. Thus each user may have individual views and navigational possibilities for working with the system. From the technology perspective, new technologies emerge and older ones fade out. If a system has a too tight coupling with some technology, it may become obsolete with the change in technology. A layered approach that provides some extent of separation from the technology is more suitable, making the overall structure still working if there is a change in the technology or even in case of replacement by the newer ones [6].

## 4. System Architecture

### 4.1 Data acquisition

Currently web-applications and frameworks are not designed to deal with semantic issues as described in the previous section, as mainly human users are analyzing and interpreting the data themselves. However, this makes the process of solving more complex problems which requires aggregated information from various sources time consuming, inconsistent, unreliable and hence inconvenient. So the need for a well-defined interface to data repositories on the web as well as personal data stores is undeniable [12].

Therefore, all information entities associated with one's lifespan must be stored in an ontological way according to some already established metadata frameworks such as RDF and Topic Maps, to facilitate the semantic queries, life trails and processing of life events. Information items can be of various kinds such as documents, emails, images, audio or video streams.

Hence the first step in creating a 'SemanticLIFE' repository is to implement a powerful data acquisition module. Basically three different types of data sources are distinguished:

- Data acquired automatically and stored in a semantic data store
- Data acquired or enriched manually by the user
- External data sources that are invoked when needed, and are not imported into the semantic data store

The third type is—in the strict sense—no data acquisition step. But it is important to understand, that there are data repositories that are not reasonable to import into the system, e.g., typically because they are external, fast changing, contain huge amounts of data and are highly structured by definition. Examples could be literature databases, enterprise information systems, company databases, web-search engines and so on. These external sources are invoked on demand (query) and a fitting ontological representation is generated by a plug-in defined in the system (see Figure 1).

Data import (in the first two cases) is performed using a standardized interface that encapsulates the data chunks into a (XML) message. This message oriented design (MOD) has the advantage of loose coupling, which means, that various modules (as described below) can be easily connected and controlled using a central message queue and event handler. Besides, the message queue performs standard operations like adding time-stamps at specific operations and creates logging mechanisms, which allow analyzing the behavior of the system in case of problems. Moreover, the usage of MOD allows future enhancements to guarantee scalability and flexibility.

### 4.2 Analysis Module

In order to efficiently extract the meta-information, the incoming messages are sent to the analysis interface in the first step. This is basically a plug-in mechanism that allows adding various analysis modules to pre-process messages of certain types. In these analysis steps meta-data is generated and added to the message. Depending on the message type, more then one analysis module might be invoked in processing a particular message. But it is important to understand at that point, that no data is removed during these analysis steps. This is desired to guarantee that no original data is lost or modified, and the history of changes is preserved! Moreover it allows re-processing data already in the semantic store in the case, that more powerful analysis modules are available in the future.

As mentioned above, incoming messages may contain the information items, placed in a nested manner. Additionally, an information item can have some other information items attached with it, like an email with attachments. Also, an information item can have other information items embedded within, like images, audio or video clips within an HTML or PDF file. Consequently, all the information items have to be analyzed in a nested way for the extraction of meta-information available in their respective headers.

Of course, currently automatic information extraction is limited to messages or data sources that provide certain machine-readable structures. For example, it is still extremely difficult to extract structured information (description) about the content of a picture or a movie. Making manual annotations to information items being fed into the system will ultimately improve its quality. This should act as a complement to content analysis and automatic metadata extraction described earlier.

After all analysis steps are finished the original message, along with the generated metadata is sent back to the message queue, which forwards it to the storage module.

## 4.3 Indexing and Storage

The Storage or Write module is responsible for extracting the XML-based metadata information from the message and writes it into the semantic data repository. Upon reception of a message from the Event Handler, the following steps are performed:

- Parsing of the message and extraction of metadata
- Invoke the appropriate plug-in for writing to specific ontology framework, via Connector Interface
- Updating indices

A basic consideration regarding the storage is, that the size of storage media is continuously increasing, which makes it possible to keep all data that were entered into the system. This is a very basic feature of the system, as it allows keeping a personal history from the data as well as from the semantic viewpoint.

Besides annotating entities with metadata, building connections with related entities is a crucial feature of the storage module. Such relationships can be built automatically or with human intervention leading to the concept of weak and strong links respectively. The weak link creation could be carried out periodically or based upon some related events. For example, email objects can be related based on the senders, the receivers or automatically by the subjects. But sometimes also relations based on some criteria given by the user are needed. Later, the user can retrieve information along with other related information. Then it would be possible to retrieve a trail of documents, messages and pictures [11].

The connections stored in the system should also have the ability of refreshing or updating itself in an automatic or semi-automatic manner. The updates occur as a result of changes in instance values, class hierarchy such as relations of classes and subclasses, kinds of properties in ontology and rules in the knowledge base.

Additionally, this message/event-oriented architecture allows solving problems for which typically active database systems are used. Trigger mechanisms in the database might react due to user-defined rules and send notification to clients.

## 4.4 Ontology Repository Standards

OWL and Topic Maps are two competing technologies for creating ontologies. The former is progressively emerged from XML, RDF, and RDFS by the W3C, while the later is specified by ISO / IEC, and is based upon SGML, XML [3].

Both can be used equally for ontology repositories but with a different level of abstraction. Topic Maps are stimulated by the proven notion of book indices, thesauri and glossaries. It is the knowledge representation applied to information management from the perspective of humans. Though, easy to perceive at the beginning, it could turn out during further research, that it might limit the ways for knowledge representation. RDF is knowledge representation applied to information management from the perspective of machines [10]. It offers low level concepts and is resource oriented.

Interoperability between the two is possible [16]. In Topic Maps, the notion of PSI (Published Subject Indicator) promotes interoperability across applications. Since, PSIs are based upon URIs, therefore, interoperability across RDF-based applications should not pose a problem.

Consequently our team is implementing multiple prototypes following the described architecture applied to specific ontological problems using both standards. Future evaluation will show, which technology will be used for the final version of the system.

### 4.5 Interactive Information Retrieval: Search and Query Functionality

Structured database systems like relational or object oriented database systems usually provide *query* mechanisms that allow powerful queries on highly structured data. As mentioned above, it is difficult to define highly structured queries (e.g., in a language like SQL) when a multitude of information systems are addressed or the information is only semi-structured. Hence the system must be capable to work with 'weaker' *search* terminology that has to be transformed into more specific queries by the system.

As the data is already stored semantically enriched (or the metadata is added 'on the fly' invoking external data sources respectively), it is possible to provide more powerful 'imprecise searches', that go far beyond 'simple' full-text indices and return information to the user in more meaningful, rich and intuitive ways. But the term 'imprecise' has two meanings: firstly, the generated queries are about undefined targets. Secondly, the target of the query is specified but there is ambiguity in the query. Therefore, the system has to solve these problems during query generation, by exploring the system's database and ontology repository and generate queries for a specific technology.

Finally these specific queries could be ontological query languages like TMQL for Topic Maps, and RQL or RDQL for RDF. For nested and distributed sub-queries, many joins and complex sub-queries would require ordering, re-ordering by the Query Optimization module, so that an optimal query execution plan is generated. Also, suitable search algorithms for non-ontological searches are to be used. Later, the generated queries could also be stored for possible reuse for system optimization.

The next step is query post processing: The received query output needs analysis and ranking to derive more precise results matched to user's requests and preferences. Analysis and ranking would rely on specific calculations in terms of semantic distances and reasoning steps to give more concrete and accurate results. To perform these tasks, the system has to refer to the set of rules and user preferences. For this purpose data from internal or external data sources are also taken into account. As the system keeps the complete history of entered data post processing needs to filter data according to the desired time-span for the problem domain.

After analysis and ranking of the query results, the system would aggregate the results before sending the result back again encapsulated in an (XML) message, which can be presented to the user through presentation application or an intermediary server, that prepares the search result, e.g., for rendering in a web-browser. The query user interface needs to be designed in a way that the user is able to write and refine the queries in an iterative manner, which is on the backend supported by the query engine of the server.

### 4.6 Extensibility and Persistence of Information and Semantic

The core of the system is based on its analysis and metadata extraction capabilities. New data sources emerge by the time and need to be treated by the system. It becomes a time consuming task to add support for newer data sources if the system has tight coupling with its components. A light weight asynchronous messaging based solution is required where new modules could be just plugged into the system without any change in the existing code.

Furthermore, openness is an extremely important issue considering systems that are designed for a lifetime acquisition of information and metadata [17]. Hence the 'SemanticLIFE' project is developed under an Open Source license, which allows various groups of developers and users to enhance and maintain the system. Moreover for each module it is evaluated if open standards exist for the representation of data, semantics and data exchange.

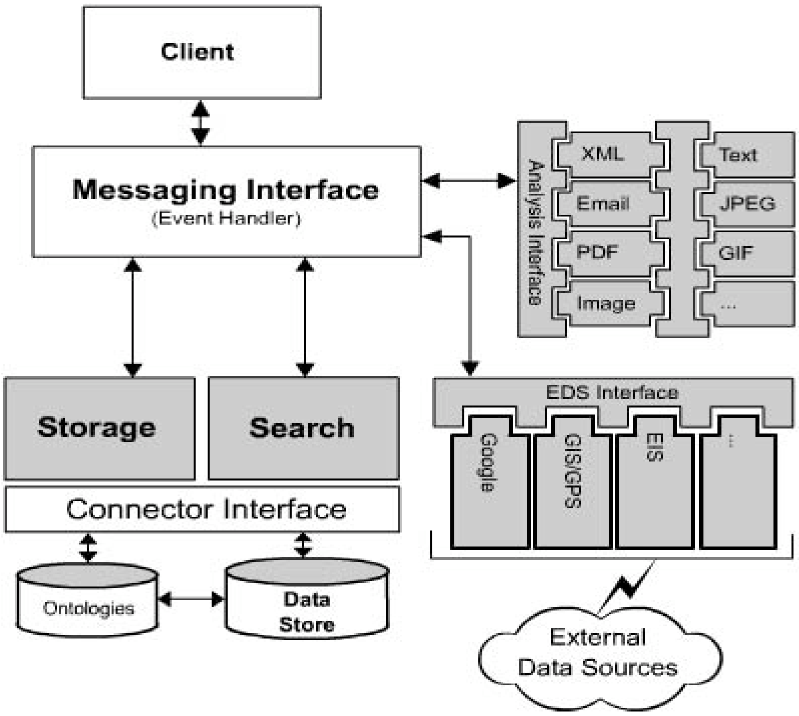

**Figure 1. Architecture for 'SemanticLIFE' Framework**

### 4.7 Client Interaction

The messaging interface provides a standardized communication mechanism for various types of clients as described above. In the current prototypical implementations, the system is oriented on individual users. However, it is clear, that future (server) versions need to be implemented as multi-user systems.

This is an encouraging task, but the arising problems are mainly ‘typical’ problems of multi-user server systems. As the focus of this paper is treatment of semantically enriched information, only a brief discussion of these issues is given here.

First of all user management and authentication requirements need to be implemented into the server system. As a consequence each message has to keep the information of the user (who performs a query for example). All sensitive processing units (mainly the storage and search module) then have to filter the messages or query results according to the user rights and roles defined for the user. This will be most probably an ontology by itself and will be stored in the server.

This user module might additionally maintain profiles of various types, such as users, devices, applications, tasks and interaction objects. This will help in presenting the results to the user in a way which is more personalized, accessible and fits the input/output capabilities of the device being used (thin, fat clients…).

## 5. Conclusion and Future Work

The ‘SemanticLIFE’ project is work in progress. Prototypes are developed to test the various aspects and design alternatives, primarily oriented for single user capabilities. The most important steps currently are to develop proof of concepts of all planned technologies limiting e.g., data feed to specific systems like mail servers.

As many standards and tools are still under heavy development, our approach was and is to develop multiple prototypes with the same functionality using different technologies (e.g., Topic Maps and RDF). In this ‘evolutionary’ approach the best strategy will succeed and implemented into the final system. Therefore, we are developing testing scenarios for a comprehensive evaluation of our prototypes.

As soon as technology and standard decisions converge, different team members will focus on specific parts of the system including multi-user and security issues, user interface design (particularly concerning queries and ontology editing) and developing testing schemes to explore the limits of the server approach, particularly concerning the amount of data, query performance and size of the ontology.

**Acknowledgements**

This work is supported by the ASEAN-EU University Network Initiative.

## 6. References

[1] BECKETT D. ed., 2004, RDF/XML Syntax Specification (Revised), http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/ (May 19, 2004)

[2] BERNERS-LEE, T., HENDLER, J., LASSILA, O., 2001, The Semantic Web, Scientific American.

[3] BIEZUNSKI, M., BRYAN, M., NEWCOMB, S. R., December 1999, ISO/IEC 13250 Topic Maps, http://www.y12.doe.gov/sgml/sc34/document/0129.pdf (April 07, 2004)

[4] DAVIS, J., FENSEL D., HARMELEN F., 2003, Towards the Semantic Web: Ontology-Driven Knowledge Management, John Wiley & Sons.

[5] DEAN M. SCHREIBER G., ed., OWL Web Ontology Language Reference, http://www.w3.org/TR/2004/REC-owl-ref-20040210/ (May 18, 2004)

[6] DERTOUZOS, M. L., 2002, The Unfinished Revolution: How to Make Technology Work for Us Instead of the Other Way Around, Harper Collins Publishers, 224 p.

[7] DOLOG, P., HENZE N., NEJDL, W., et al, 2003, Towards the Adaptive Semantic Web, Lecture Notes in Computer Science, Volume 2901, Springer-Verlag Heidelberg.

[8] FENSEL, D., HENDLER J., LIEBERMAN H., WAHLSTER W., 2003, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential, The MIT Press.

[9] FREEMAN, E., GELERNTER, D., 1996, LifeStreams: A storage model for Personal Data, ACM SIGMOD Record.

[10] GARSHOL, L. M., 2003, Living with Topic Maps and RDF, Ontopia Technical Report, http://www.ontopia.net/topicmaps/materials/tmrdf.html (March 24, 2004)

[11] GEMMEL, J., BELL, G., LUEDER, R., DRUCKER, S., WONG, C., 2002, MyLifeBits: Fulfilling the Memex Vision, Proceedings of ACM Multimedia '02, ACM Press, p. 235-238.

[12] GEROIMENKO V., CHEN C., 2003, Visualizing the Semantic Web: XML-based Internet and Information Visualization, Springer-Verlag, London, 202 p.

[13] HORROCKS I., PATEL-SCHNEIDER, P. F., VAN HARMELEN F., ZIENBERER, JASON, Y., 2003, From SHIQ and RDF to OWL: The Making of a Web Ontology Language, Elsevier's Journal of Web Semantics, Volume 1, Issue 1.

[14] HUYNH D., KARGER, D., QUAN D., 2002, Haystack: A Platform for Creating, Organizing and Visualizing Information Using RDF, Semantic Web Workshop.

[15] IBIDUNNI, O., 2002, Supporting Workgroups Collaborating via Email Using the Semantic Web and RDF, HP Labs, Technical Report HPL-2002-316, http://www.hpl.hp.com/techreports/2002/HPL-2002-316.html (May 14, 2004)

[16] LACHER, M. S., DECKER, S., 2001, On the Integration of Topic Maps and RDF Data, Proceedings of SWWS '01, California.

[17] MCBRIDE B., 2002, Four Steps towards the Widespread Adoption of a Semantic Web, First International Semantic Web Conference, Sardinia.

[18] NICOLAU J.M., 1996, On Thoughts About The Brain, in JOSEFMORENO-DIAZ, R., MIRA-MIRA, J. ed., Brain Processes, Theories and Models, The MIT Press, p. 71-77.

[19] NEJDL, W., WOLF, B., STAAB, S., TANE, J., et al, 2002, EDUTELLA: Searching an Annotating Resources within an RDF-based P2P Networks, The 11th International WWW Conference, Hawaii.

[20] REYNOLDS D., e-Person - Personal Information Infrastructure, HP Labs, http://www.hpl.hp.com/semweb/e-person.htm (January 08, 2004)

[21] Semagix Freedom, http://www.semagix.com (April 13, 2004)