



EKA 2010 • Workshop W2

Monday • 11th october 2010

Personal Semantic Data

*Laura Dragan, Bernhard Schandl, Charlie Abela, Tudor Groza,
Gunnar Aastrand Grimnes, Stefan Decker*

Preface

Welcome to the first workshop on Personal Semantic Data (PSD2010), part of the 17th International Conference on Knowledge Engineering and Knowledge Management (EKAW2010)!

Personal information management (PIM) is an active area of interest for research and industry alike. While our time and energy resources remain constant, the amount of information that needs our attention grows exponentially with the advances in communications and information sharing tools.

The tools that we use to manage our personal information have evolved over time from the pen and paper day planners to their numerous digital replacements. The desktop used to be at the centre of the users' PIM universe, containing their contacts, emails, events, appointments, and to-do lists. However, as the amount of stored information and the number of applications available to handle it grew, desktop data became harder and harder to manage, as it was locked-in by applications and stored in application-specific formats. The Semantic Desktop is the result of applying Semantic Web technologies to the desktop, to better interlink personal data and make it easier to search, browse and organise. It lifted the data from the application silos and non-standard formats to a standard RDF-based representation, described using commonly agreed-upon ontologies.

Nowadays, the transition is made more and more towards mobile devices, the majority of which have Internet connectivity. This has led to an increasing share of information, like calendar and email, being stored on users' various devices or in the cloud, because of hardware limitations like storage and processing power. Also, applications such as Chrome OS, Google Documents, or MS Office Live enable users to store personal documents in the Cloud, while many social relations are managed through social Web sites like Facebook, MySpace or Bebo. In parallel, the Semantic Web has gained considerable momentum, especially through initiatives like Linking Open Data, that have generated a vast amount of structured data available on the Web. Furthermore, projects like FOAF and SIOC have enabled the publication of machine-readable information about people and their social interactions.

As more online services and applications become available to users and gain popularity, the boundaries between the desktop and the Web become less discernible. The desktop is no longer the single access point to personal information, but one of many personal information sources. Consequently, personal information is becoming more fragmented across multiple devices, requiring extra effort to synchronize, duplicate, search and browse. We believe that semantic technologies can improve significantly the user's experience and relieve some of the stress associated with managing disparate information.

Personal semantic data is scattered over several media, and while semantic technologies are already successfully deployed on the Web as well as on the

desktop, data integration is not always straightforward. The transition from the desktop to a distributed system for PIM raises new challenges, which represent the subject of this workshop. Related research is being conducted in several disciplines like human-computer interaction, privacy and security, information extraction and matching. Through this workshop we would like to enable cross-domain collaborations to further advance the use of technologies from the Semantic Web and the Web of Data for Personal Information Management, and to explore and discuss approaches for improving PIM through the use of vast amounts of (semantic) information available online. In turn, this workshop is of interest to researchers in the areas of PIM, Linked Data, Web Sciences, Social Collaboration, and more.

We wish to thank all the authors of submitted papers and to the members of the program committee.

October 2010

The organizers

Organization Committee

Laura Drăgan :

Affiliation: Digital Enterprise Research Institute (DERI), National University of Ireland, Galway

Email: laura.dragan@deri.org

Web page: http://www.deri.ie/about/team/member/laura_dragan/

Bernhard Schandl :

Affiliation: Department of Distributed and Multimedia Systems, University of Vienna, Austria

Email: bernhard.schandl@univie.ac.at

Web page: <http://www.cs.univie.ac.at/bernhard.schandl>

Charlie Abela :

Affiliation: Department of Intelligent Computer Systems (ICS), University of Malta, Malta

Email: charlie.abela@um.edu.mt

Web page: <http://staff.um.edu.mt/cabe2/>

Tudor Groza :

Affiliation: Digital Enterprise Research Institute (DERI), National University of Ireland, Galway

Email: tudor.groza@deri.org

Web page: <http://www.tudorgroza.org>

Gunnar Aastrand Grimnes :

Affiliation: DFKI GmbH, Germany

Email: gunnar.grimnes@dfki.de

Web page: <http://www.dfki.uni-kl.de/~grimnes/>

Prof. Stefan Decker :

Affiliation: Digital Enterprise Research Institute (DERI), National University of Ireland, Galway

Email: stefan.decker@deri.org

Web page: <http://www.stefandecker.org>

Program Committee

Diego Berrueta, *CTIC Foundation, Gijon, Spain*
Dan Brickley, *FOAF Project, UK*
François Bry, *Ludwig-Maximilian University Munich, Germany*
Jerome Euzenat, *INRIA Grenoble Rhone-Alpes, France*
Fabien Gandon, *INRIA Sophia-Antipolis, France*
Harry Halpin, *University of Edinburgh, UK*
Nicola Henze, *Leibniz University Hannover, Germany*
Robert Jaeschke, *University of Kassel, Germany*
William Jones, *The Information School, University of Washington, USA*
Malte Kiesel, *DFKI GmbH, Germany*
Stéphane Laurière, *Mandriva, France*
Knud Möller, *Digital Enterprise Research Institute (DERI), Galway, Ireland*
Paola Monachesi, *Utrecht University, Utrecht, The Netherlands*
Daniel Olmedilla, *Telefonica R & D, Spain*
Gerald Reif, *University of Zurich, Department of Informatics, Switzerland*
Leo Sauermann, *gnowsis.com, Vienna, Austria*
Sven Schwarz, *DFKI GmbH, Germany*
Chris Staff, *Department of Intelligent Computer Systems, University of Malta*
Diman Todorov, *Knowledge Engineering Systems Group, Cardiff University, UK*
Mischa Tuffield, *Garlik, UK*
Claudia Wagner, *TU Graz, Austria*
Stefan Zander, *Department of Distributed and Multimedia Systems, University of Vienna, Austria*

Copyright remains with the authors, and permission to reproduce material printed here should be sought from them. Similarly, pursuing copyright infringements, plagiarism, etc. remains the responsibility of authors.

Table of Contents

Keynote

Making Sense of Users' Web Activity	1
<i>Mathieu D'Aquin</i>	

Full Papers

Managing Personal Information by Automatic Titling of E-mails	2
<i>Cédric Lopez, Violaine Prince, Mathieu Roche</i>	
SemChat: Extracting Personal Information from Chat Conversations	14
<i>Keith Cortis, Charlie Abela</i>	
Ad-hoc File Sharing Using Linked Data Technologies	26
<i>Niko Popitsch, Bernhard Schandl</i>	
Towards a Simple Textual Trace Based Personal Exo-Memory	38
<i>Pierre Deransart</i>	

Short Paper

LinksTo - A Web2.0 System that Utilises Linked Data Principles to Link Related Resources Together	50
<i>Owen Sacco, Matthew Montebello</i>	

Making Sense of Users' Web Activity

Mathieu d'Aquin

Knowledge Media Institute, The Open University, Milton Keynes, UK
{m.daquin}@open.ac.uk

Personal information management (PIM), as described by [1], is “*the practice and study of the activities people perform to acquire, organise, maintain, retrieve, use, and control distribution of information items*”. More and more services rely on the Web to communicate with their users. The way users can control the distribution of personal information exchanged daily through various Web channels therefore appears as a crucial task for PIM. However, while the definition above clearly covers such activities, PIM has traditionally been focusing more on the aspects of supporting information organisation and integration for the purpose retrieval. Indeed, the types of personal information mentioned in [1] include elements such as “*information about a person but kept by and under the control of others*”, but ignore one of the most difficult type of information to manage: *information about a person which is being shared and exposed to others*.

The related issues not only concern the ways to monitor, store and retrieve this specific type of information, but also the ways for users to make sense of the huge amounts of information they are exchanging on the Web, knowingly or unknowingly. Indeed, as a first building block in this area, we developed a tool dedicated to tracking the activity of an individual user on the Web. In practice, this tool takes the form of a ‘local proxy’ intercepting and storing (using Semantic Web standards) the HTTP traffic on the user’s computer. At a higher level, we can see this tool as a ‘Web Liffellogger’, dedicated to the indiscriminating collection of information concerning the user’s online activity. While relatively basic in principle, experimenting with this tool over a period of time generates huge amounts of data (100 Million Triples for a single user in 2.5 months) which, when studied, allows us to unveil interesting, and sometimes surprising aspects of the users Web life.

The use of semantic technologies offers the right level of flexibility for the management of such large, heterogeneous data, but more importantly, provides us with the data integration and modelling approaches necessary to making sense of the data. For example, mapping the collected semantic logs with a representation of the user profile allows us to construct models of the perceived trust the user gives to various websites regarding the handling of his/her personal information, and of the sensitivity of this information. Going a step further, by applying different ontologies over the data, and linking it to the Web of Data, we can build different perspectives on the traces of Web activity produced by the user, providing as many “interpretations” of the user’s interaction with the Web, in addition to tools supporting him/her in managing this interaction.

1. William Jones and Jaime Teevan (editors), *Personal Information Management*, University of Washington Press, 2007

Managing Personal Information by Automatic Titling of E-mails

Cédric Lopez, Violaine Prince, and Mathieu Roche

Univ. Montpellier 2, LIRMM, Montpellier, France
{lopez,prince,mroche}@lirmm.fr,
WWW home page: <http://www.lirmm.fr/>

Abstract. This paper presents an approach that enables automatic titling of e-mails relying on the morphosyntactic study of real titles. Automatic titling of e-mails has two interests: Titling mails 'no object' and managing personal information. The method is developed in three stages: Candidate sentences determination for titling, noun phrases extraction in the candidate sentences, and finally, selecting a particular noun phrase as a possible e-mail title. A human evaluation associated with ROC Curves are presented.

1 Introduction

A title definition met in any dictionary is 'word, expression, sentence, etc., serving to indicate a paper, one of its parts [...], to give its subject.' So it seems that a title role can be assumed by a well formed word group, an expression, a topic or a simple word, related to the text content, in one way or another. It ensues that some groups of well formed words can be convenient for a title, which means that a text might get several possible titles. A title varies in length (i.e. number of words), form and local focus. So, the human judgment on a title quality will always be subjective and several different titles might be judged as relevant to a given content.

This paper deals with an automatic approach providing a title to an e-mail, which meets the different characteristics of human issued titles. So, when a title is absent (e-mails without subject), the described method enables the user to save time by informing him/her in order to manage its personal data. Actually, a relevant title is an important issue for the person who wants to correctly classify its e-mails. Let us note that titling is not a task to be confused with automatic summarization, text compression, and indexation, although it has several common points with them. This will be detailed in the 'related work' section.

The originality of this method is that it relies on the morphosyntactic characteristics of existing titles to automatically generate a document heading. So the first step is to determine the nature of the morphosyntactic structure in e-mail titles. A basic hunch is that a key term of a text can be used as its title. But studies have shown that very few titles are restricted to a single term.

Besides, the reformulation of a text relevant elements is still a quite difficult task, which will not be addressed in the present work. The state-of-the art in automatic titling (section 2) and our own corpus study have stressed out the following hypothesis: It seems that the first sentences of a document tend to contain the relevant information for a possible title. Our approach (section 3) extracts crucial knowledge in these selected sentences and provide a title. An evaluation obtained on real data is presented in section 4.

2 Related Work

It seems that no scientific study leading to an automatic titling application was published. However, the title issue is studied in numerous works.

Titling is a process aiming at relevantly representing the contents of documents. It might use metaphors, humor or emphasis, thus separating a titling task from a summarization process, proving the importance of rhetorical status in both tasks [13]. Titles have been studied as textual objects focusing on fonts, sizes, colors, . . . [6]. Also, since a title suggests an outline of the associated document topic, it is endowed with a semantic contents that has three functions: Interest and captivate the reader, inform the reader, introduce the topic of the text.

It was noticed that elements appearing in the title are often present in the body of the text [18]. [1] has showed that the first and last sentences of paragraphs are considered important. The recent work of [2, 7, 19] supports this idea and shows that the covering rate of those words present in titles, is very high in the first sentences of a text. [14] notices that very often, a definition is given in the first sentences following the title, especially in informative or academic texts, meaning that relevant words tend to appear in the beginning since definitions introduce the text subject while exhibiting its complex terms. The latter indicate relevant semantic entities and constitute a better representation of the semantic document contents [10].

A title is not exactly the smallest possible abstract. While a summary, the most condensed form of a text, has to give an outline of the text contents that respects the text structure, a title indicates the treated subject in the text without revealing all the content [15]. Summarization might rely on titles, such as in [5] where titles are systematically used to create the summary. This method stresses out the title role, but also the necessity to know the title to obtain a good summary. Text compression could be interesting for titling if a strong compression could be undertaken, resulting in a single relevant word group. Compression texts methods (e.g. [17]) could be used to choose a word group obeying to titles constraints. However, one has to largely prune compression results to select the relevant group [13].

A title is not an index: A title does not necessarily contain key words (and indexes are key words), and might present a partial or total reformulation of the text (what an index is not).

Finally, a title is a full entity, has its own functions, and titling has to be sharply distinguished from summarizing and indexing.

A rapid survey of existing documents helps to fathom some of title characteristics such as length, and nature of part-of-speech items often used. Next section is devoted to our automatic titling approach.

3 The Automatic Titling Approach

By leaning on the previous work (section 2) and our previous study [9], we propose an automatic titling approach in order to title e-mails.

The first elementary step consists in determining the textual data from which we will build a title. These data have to contain the information necessary for the titling of the document. As said before, [6] has concluded that the maximal covering of the words of the title in the text, was obtained by extracting the first seven sentences and both last ones.

The following sections present our methods. The main idea consists in selecting the most relevant Noun Phrase (NP) for its use as title [8].

3.1 Extracting of the Noun Phrases (NP)

Corpus analysis showed that the titles of e-mails contain few verbs and are short (between approximately two and six words) (Table 1). Our aim is to extract the most relevant noun phrases in order to provide a title.

Nature	% Noun	% Named entity	% Verb	Number of Words
E-mails	73	53	6	5

Table 1. Statistics on real titles of our corpus

For that purpose, e-mails are tagged with TreeTagger [12]. Our NP extraction method is inspired from [3] who determined syntactical patterns allowing noun phrase extraction, e.g. *Noun1 – Adjective1*, *Noun1 – Det1 – Noun2*, *Noun1 – Noun2*, and so forth. We set up syntactical filters, adapted to French, allowing the extraction of NP having a maximal size of 6 words (For example 'noun - prep - det - noun - prep - det'). This limit of size is inspired from the maximal title length for e-mails.

Next section consist in selecting the most relevant NP extracted, for its use as title. In the following section, we shall use the TF-IDF measure to calculate the score of every NP. This score can be the maximal TF-IDF obtained for a word of the NP (T_{MAX}) either the sum of the TF-IDF of every word of the NP (T_{SUM}). Finally, the T_{ALL} method is presented.

3.2 Selection of NP with statistical criteria

We shall use the TF-IDF measure [11] to calculate the score of every NP extracted from the e-mail text.

The TF-IDF measure is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in the corpus.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k (n_{k,j})} \quad (1)$$

$n_{i,j}$ is the number of occurrences of the considered term t_i in document d_j , and the denominator is the sum of number of occurrences of all terms in document d_j .

$$idf_i = \log \frac{|D|}{|d_j : t_i \in d_j|} \quad (2)$$

$|D|$: total number of documents in the corpus.

$|d_j : t_i \in d_j|$: number of documents where the term t_i appears.

Let us note that if new emails arrive in the corpus, the TF-IDF will be recalculated. The NP score can be the maximal TF-IDF obtained for a word of the NP (T_{MAX}) either the sum of the TF-IDF of every word of the NP (T_{SUM}). Finally, an improvement of these methods is presented (T_{ALL}).

T_{MAX} . The T_{MAX} method consists in calculating a score for each NP in the first sentences [6]. For each word of the candidate NP, the TF-IDF is computed. The score for each candidate NP is the maximum TF-IDF of the words of the NP. With this method, discriminant terms are highlighted. For example, in the noun phrase 'contribution recherche' (*research contribution*) ($NP1$) and 'nouvelle relecture' (*new review*) ($NP2$), $NP1$ will be retained, the term *contribution* being more discriminant than 'recherche' (*research*), 'nouvelle' (*new*), and 'relecture' (*review*) in our e-mail corpus.

Contrarily to T_{MAX} , another method consists in extracting the NP containing the most information: T_{SUM} .

T_{SUM} . For each word of the NP candidate (extracted from first sentences of the e-mail), the TF-IDF is calculated. The score of each NP candidate is the sum of each TF-IDF. This method favors long noun phrases. For example, let both NP 'souis de vibration' (*vibration nuisance*) ($NP3$) and 'souis de vibration avec Saxo' (*vibration nuisance with Saxo*) ($NP4$). $NP4$ will be privileged because it is a superset of $NP3$. However, this method still allows to distinguish between noun phrases of the same size: $NP2$ obtains a better score than $NP1$ because

the sum of the TF-IDF for the terms 'nouvelle' (*new*) and 'relecture' (*review*) is higher than the sum for 'contribution' (*contribution*) and 'recherche' (*research*).

With these methods (T_{MAX} and T_{SUM}), we only worked on the first sentences (two sentences) of the e-mails. In the next section, we propose an approach using all the texts.

T_{ALL} . Generally, it is advisable that relevant terms for titling are present in the first and last sentences of the text (see Section 2). However, as regards e-mails, our statistic study shows that terms appearing in real title are rarely at the end of the text (Fig. 1).

In the Figure 1, the Y axis represents the number of words that appears both in the title and in the text. The X axis represents the parts of the text. Actually in order to identify the parts of the text where the terms of the title appear, the text was divided in eight parts. For instance, in the Figure 1, four words are both in the title and on the sixth part of the text. Of course, determiners, prepositions, articles, and so forth, are not considered in this study. We note that the dispersal of relevant terms in the text takes an hyperbolic form.

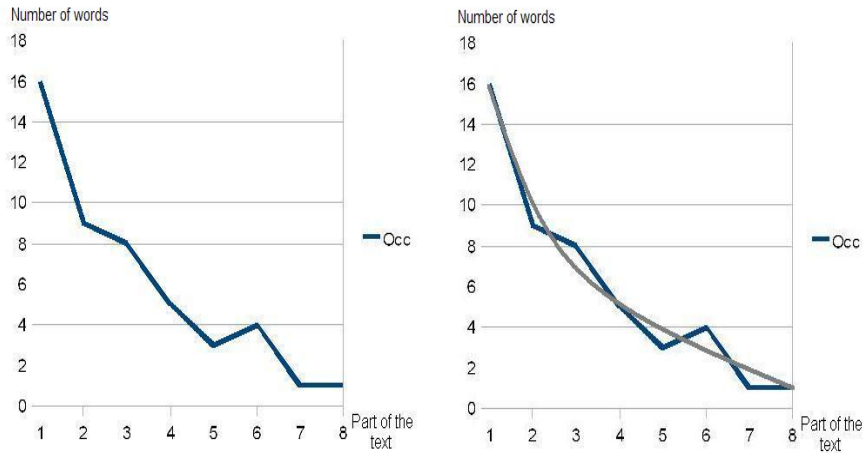


Fig. 1. Covering rate of words of text appearing in real titles, and median curve (based on the 30 last personal e-mails received).

Let us note that if the NP score is based only on the TF-IDF ¹, the results indicate that NP candidates for a title could be extracted wherever in the text

¹ Score calculated in the same way as T_{SUM} , but on the complete text and not only on the first sentences

(Fig. 2). We will see that this method, called T_{FREQ} , does not obtain good results (see Section 4).

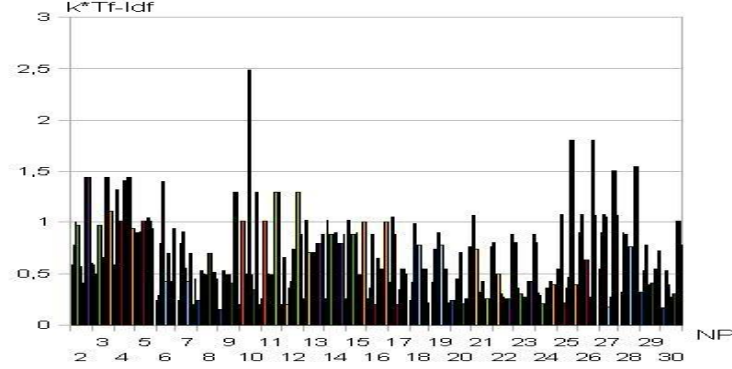


Fig. 2. Dispersal of NP, with a TF-IDF score (with k coefficient).

Our objective is to use this information during the calculation of the NP score. We propose a method combining the NP position in the text and its semantic contents.

The $Score_P$ enables to give more importance to the NP extracted at the beginning (section 3.2) of the text. P is the position of the NP (e.g., 1 for $NP_{number 1}$, 43 for $NP_{number 43}$). We use $\alpha = \frac{1}{2}$. In a future work, we plan to apply different values to α .

$$Score_P = \frac{1}{P^\alpha} \quad (3)$$

The $Score_{TF-IDF}$ is calculated in the same way as T_{SUM} , but on the complete text and not only on the first two sentences. Finally, the score of the NP ($Score_{T_{ALL}}$) is the sum of $Score_P$ and $Score_{TF-IDF}$.

$$Score_{TF-IDF} = \sum_{term=1}^n (TF * IDF)_{term} \quad (4)$$

$$Score_{T_{ALL}} = Score_P + Score_{TF-IDF} \quad (5)$$

With the example given in the Fig. 3, the fourth extracted NP is chosen:

1. Dans un soucis (In a concern)
2. Soucis d'amélioration (Concerns of improvements)
3. Amélioration de la Journée (Improvement of the Day)

Bonjour,

Dans un souci d'amélioration de la Journée Scientifique du LIRMM (que nous souhaitons pérenniser à la fréquence d'une fois par an), pouvez-vous me faire parvenir vos suggestions/remarques concernant :

- la qualité du programme,
- les exposés auxquels vous avez pu assister (pas assez/trop longs, pas assez/trop vulgarisés, etc.)
- les contenus scientifiques que vous auriez aimé voir au programme d'une telle journée,
- l'organisation de la journée,
- ...autre sujet ?

N'hésitez pas à me faire parvenir vos remarques, qu'elles soient positives ou négatives, même si vous n'avez pas participé (avec les raisons de cette non-participation).

Merci,

Caroline

Fig. 3. E-mail example.

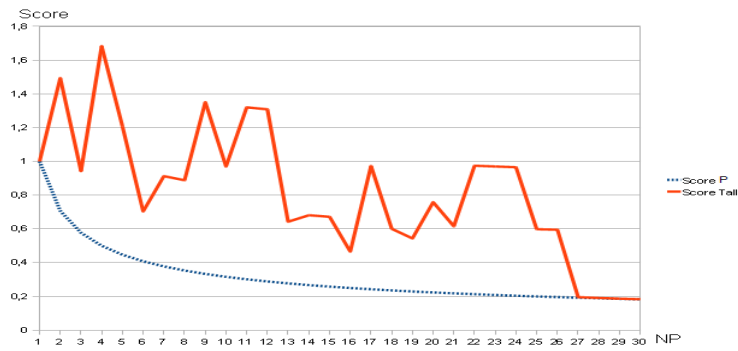


Fig. 4. Representation of $Score_P$ and $Score_{TALL}$ curves for an e-mail.

4. *Amélioration de la Journée Scientifique (Improvement of the Scientific Day)*
5. La Journée Scientifique du LIRMM (The Scientific Day of the LIRMM)
6. Scientifique du LIRMM (Scientific of the LIRMM)
7. Du LIRMM (Of the LIRMM)
8. LIRMM
9. La fréquence d'une fois (Frequency of one time)
10. ...

The Figure 4 shows that the $Score_P$ gives an important weight to the first noun phrases. Moreover, the second and fourth NP have an important value of $Score_{TF-IDF}$. Finally, the $Score_{T_{ALL}}$ favors the fourth NP as a relevant title.

4 Experiments

The corpora consists of French personal e-mails from different persons and registers ; they are more or less well written. Our three methods studied in this paper are evaluated. First of all, we have studied the behavior of our methods by using ROC Curves.

4.1 ROC Curves

ROC Curves measure the quality of the obtained ranking. Initially the ROC Curves (Receiver Operating Characteristic), detailed in [4], come from the field of signal processing. ROC Curves are often used in medicine to evaluate the validity of diagnosis tests. ROC Curves show in X-coordinate the rate of false positives (in our case, not relevant title) and in Y-coordinate the rate of true positives (relevant titles). The surface under the ROC Curve (*AUC - Area Under the Curve*), can be seen as the effectiveness of a measurement of interest. The criterion related to the surface under the curve is equivalent to the statistical test of Wilcoxon-Mann-Whitney (see [16]).

In the case of the noun phrase extracting, a perfect ROC Curve corresponds to obtaining all relevant NP at the beginning of the list and all irrelevant NP at the end of the list. This situation corresponds to $AUC = 1$.

The diagonal corresponds to the performance of a random system, progress of the rate of true positives being accompanied by an equivalent degradation of the rate of false positives. This situation corresponds to $AUC = 0.5$.

A human expert have manually evaluated the list of extracted NP, from 7 e-mails (i.e. approximately 210 NP).

ROC curves indicate that the favorable titling methods are T_{ALL} (0.77) and T_{SUM} (0.69) (see Table 2). The score of T_{ALL} (i.e. NP extracted on the whole text) seems to give better results than T_{SUM} . With T_{MAX} , the choice of the title among the NP candidate is irrelevant for e-mails.

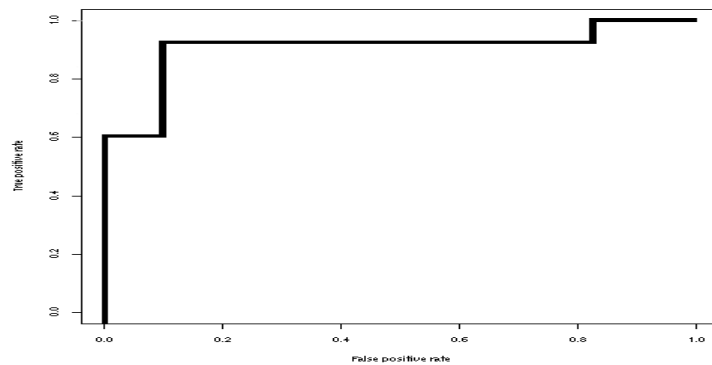


Fig. 5. Example of E-mail ROC Curve for the T_{ALL} method.

E-mails	T_{MAX}	T_{SUM}	T_{ALL}
1	0.13	0.63	0.92
2	0.08	0.5	0.96
3	0.63	0.67	0.5
4	1	1	1
5	0.23	0.21	0.62
6	0.37	0.83	0.72
7	0.75	1	0.67
AUC Avg.	0.35	0.69	0.77

Table 2. AUC Average for each method, results of ROC Curves.

4.2 Human evaluation

The experiments have been run on personal e-mails. Twenty e-mails were selected. Texts are variable in size (i.e. number of words), topics, technicality, and effort of writing. Evaluation results are presented in Table 3. The expert had to tag "−" or "+" all the titles proposed with our system. The + symbol indicates that the title given by the method (i.e. T_{MAX} , T_{FREQ} , T_{SUM} , T_{ALL}) is relevant, and − indicates a title as irrelevant.

Titling with T_{MAX} does not offer good results (9/20) perhaps because of the rarity/specificity of the terms of the title. Moreover, it could be interesting to evaluate this method on specific e-mails, for example on e-mails sent between specialists of a same domain.

Titles determined by T_{SUM} are relevant (12/20). However, the results show that any titles are irrelevant, and thus that it is possible that the titles were not found in the first two sentences.

Finally, T_{ALL} obtains a high score (16/20), that indicates a real interest to extract the NP in the whole text, with the condition of use their position. In order to see if this condition is really necessary, we have evaluated the T_{FREQ} method. This one is identical in T_{ALL} , but without the consideration of $Score_{TF-IDF}$ in the final NP score. T_{FREQ} obtains a bad result (8/20). This result justifies the use of the position score called $Score_P$ (see Section 3.2).

E-mails	T_{MAX}	T_{SUM}	T_{FREQ}	T_{ALL}
1	-	+	-	+
2	+	-	-	+
3	+	+	-	+
4	+	-	-	+
5	+	+	+	+
6	-	-	-	+
7	-	+	+	+
8	-	+	+	+
9	-	-	-	-
10	+	+	-	-
11	+	+	+	+
12	+	-	-	+
13	-	+	-	+
14	-	-	-	+
15	-	-	+	+
16	-	-	-	-
17	-	+	+	-
18	+	+	+	+
19	-	+	+	+
20	+	+	-	+
Total	9	12	8	16

Table 3. Evaluation obtained on real data (20 e-mails).

5 Conclusion

We set up a method that enables to combine the NP position importance in e-mails and its semantic content.

Statistic study shows that it is necessary to use all the sentences of the e-mail in order to propose a relevant title. The method T_{ALL} seems to be adapted to e-mails titling.

The quality of automatically computed titles strongly depends on the care brought to the text writing. Nevertheless, the T_{ALL} method² proposes relevant titles for e-mails. The results show all the same that improvements can be brought. Even if a part of the performance of this approach depends on Tree Tagger, it seems possible to improve results. In particular, it could be interesting to give more importance to Named Entities using T_{ALL} approach.

The evaluation tends to indicate a possible benefit of an automatic method. This one enables a time saving procedure for an e-mail writer... Then, the proposed title makes possible a relevant indexing process of personal data as e-mails.

References

1. Baxendale, B.: Man-made index for technical literature - an experiment. IBM Journal of Research and Development pp. 354–361 (1958)
2. Belhaoues, M.: Titrage automatique de pages web. Master Thesis, University Montpellier II, France (2009)
3. Daille, B.: Study and implementation of combined techniques for automatic extraction of terminology. The Balancing Act : Combining Symbolic and Statistical Approaches to language pp. 29–36 (1996)
4. Ferri, C., Flach, P., Hernandez-Orallo, J.: Learning decision trees using the area under the ROC curve. In: Proceedings of ICML'02. pp. 139–146 (2002)
5. Goldsteiny, J., Kantrowitz, M., Mittal, V., Carbonelly, J.: Summarizing text documents: Sentence selection and evaluation metrics. pp. 121–128 (1999)
6. Ho-Dac, L.M., Jacques, M.P., Rebeyrolle, J.: Sur la fonction discursive des titres. S. Porhiel and D. Klingler (Eds). L'unit texte, Pleyben, Perspectives. pp. 125–152 (2004)
7. Jacques, M., Rebeyrolle, J.: Titres et structuration des documents. Actes International Symposium: Discourse and Document pp. 125–152 (2004)
8. Lopez, C., Prince, V., Roche, M.: Text titling application (demonstration session, to appear). In: Proceedings of EKAW'10 (2010)
9. Lopez, C., Prince, V., Roche, M.: Titrage automatique de documents électroniques par extraction de syntagmes nominaux. In: Acte des 21èmes Journées Francophones d'Ingénierie des Connaissances. pp. 17–28 (2010)
10. Mitra, M., Buckley, C., Singhal, A., Cardi, C.: An analysis of statistical and syntactic phrases. In: RIAO'1997 (1997)
11. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management 24 p. 513–523 (1988)

² Available on the address

http://www.lirmm.fr/~lopez/Titrage_general/TiMail.php

12. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: International Conference on New Methods in Language Processing. pp. 44–49 (1994)
13. Teufel, S., Moens, M.: Sentence extraction and rhetorical classification for flexible abstracts. In: AAAI Spring Symposium on Intelligent Text Summarisation. pp. 16–25 (2002)
14. Vinet, M.T.: L’aspet et la copule vide dans la grammaire des titres. *Persee* 100, 83–101 (1993)
15. Wang, D., Zhu, S., Li, T., Gong, Y.: Multi-document summarization using sentence-based topic models. In: ACL-IJCNLP ’09: Joint conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing. pp. 297–300 (2009)
16. Yan, L., Dodier, R., Mozer, M., Wolniewicz, R.: Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic. In: Proceedings of ICML’03. pp. 848–855 (2003)
17. Yousfi-Monod, M., Prince, V.: Sentence compression as a step in summarization or an alternative path in text shortening. In: Coling’08: International Conference on Computational Linguistics, Manchester, UK. pp. 139–142 (2008)
18. Zajic, D., Door, B., Schwarz, R.: Automatic headline generation for newspaper stories. Workshop on Text Summarization (ACL 2002 and DUC 2002 meeting on Text Summarization). Philadelphia. (2002)
19. Zhou, L., Hovy, E.: Headline summarization at isi. In: Document Understanding Conference (DUC-2003), Edmonton, Alberta, Canada. (2003)

SemChat: Extracting Personal Information from Chat Conversations

Keith Cortis, Charlie Abela

Faculty of Information and Communication Technology,
Department of Intelligent Computer Systems,
University of Malta
kcor0003@um.edu.mt, charlie.abela@um.edu.mt

Abstract. The Semantic Desktop builds over Bush's Memex vision and focuses on enhancing the personal information management (PIM) process through the integration and presentation of content found on the user's desktop. In line with the Semantic Desktop's philosophy we present SemChat, which is a semantic chat client component. We discuss how SemChat allows personal information related to persons, locations, organisations, dates and events to be extracted from chat conversations and to be integrated into the user's Personal Information Model (PIMO), with annotated events being directly exported to an event scheduler. We also discuss SemChat's search facility, which allows users to search for relevant concepts within their personal chat-information space. Furthermore we elaborate on our initial evaluation efforts which proved to be very promising.

Keywords: personal information management, social semantic desktop, personal information model, semantic chat

1 Introduction

The internet has brought about a radical change in the way people interact. Online communities have flourished, first fueled by electronic mail (e-mail), and nowadays complemented by instant messaging (IM). The advent of e-mail triggered a chain reaction that naturally resulted in the development of IM in 1993, since the former is not as immediate. For this reason IM has become very popular over recent years. Common acquaintances can communicate with each other, in real time using IM whereby messages are transferred from one user to another in a seemingly peer-to-peer manner.

However with the increase in applications that allowed these virtual online communities to flourish, came also an increase in the fragmentation of personal information. It is left up to the user to integrate and manage this disparity in personal information scraps, such that these are not forgotten or lost. In this regards, numerous tools have been developed to aid users in the management of their personal information space.

The vision behind the Semantic Desktop (SD) is precisely that of tackling the difficulties when managing personal information. It builds over Bush's Memex¹ vision and focuses on enhancing the personal information management (PIM) process through the integration and presentation of content found on the user's desktop, by using Semantic Web standards and technologies. This vision is further extended within the Social Semantic Desktop (SSD) which projects the SD into the social dimension and augments SD with facilities for information distribution and collaboration [12].

In line with the Social Semantic Desktop's philosophy our research aims at exploiting and extending NEPOMUK², a Social Semantic Desktop framework, with SemChat, a semantic chat client component. The main objectives behind SemChat include the following:

- compatibility with different chat clients
- provide for the extraction and annotation of the user-relevant concepts from a chat conversation which have not already been stored within the users Personal Information Model (PIMO)³
- provide for the identification and extraction of any events mentioned during a chat conversation, together with the option to annotate such events within an available task/event scheduler.
- provide for the persistence of any concepts that were not readily annotated by the user, for reference in future SemChat sessions
- provide for a search facility over the chat-related concepts (and events)

The rest of the paper is organized as follows. In Section 2 we highlight the main ideas behind SemChat's architecture and implementation, whilst in Section 3 we present and discuss the results obtained after an initial evaluation session. We go over some related research in Section 4 and provide some future aspirations and concluding comments in Section 5 and Section 6 respectively.

2 SemChat

In Figure 1 we present a general architecture of SemChat and its main components. The motivation behind this architecture partly came from work performed on Semanta [10] and SemNotes [3] which are applications that also exploit the ideas behind SD and SSD. The former is a semantic email component while the latter is a note-taking tool, and both integrate closely with NEPOMUK.

NEPOMUK's environment allows the user to manage all the data found on her desktop and to link the documents within the PIMO [8]. This ties perfectly with one of our main objectives within SemChat precisely that of extracting user-relevant concepts and events from chat conversations and to expose and link, this extracted knowledge, with that found on the user's desktop. In this manner, the

¹ <http://cyberartsworld.org/cpace/ht/jhup/memex.html>

² <http://nepomuk.semanticdesktop.org/>

³ <http://dev.nepomuk.semanticdesktop.org/wiki/PimoOntology>

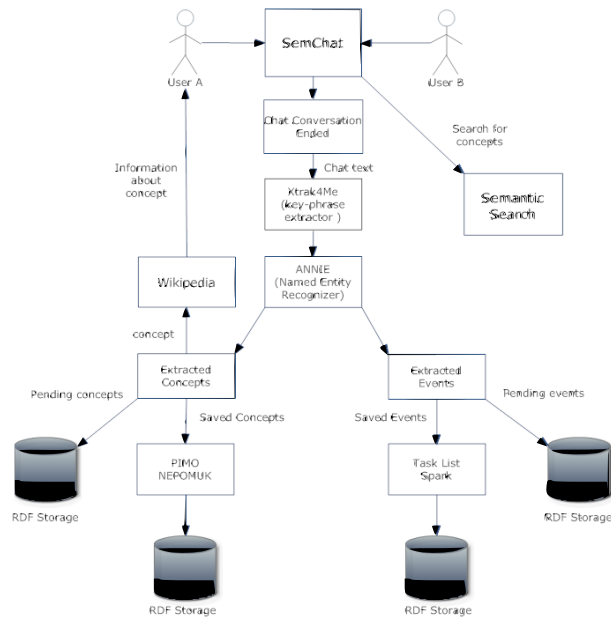


Fig. 1. General Architecture of SemChat and its main components

user's PIMO is augmented with newly found concepts mentioned during chat sessions while at the same time during conversations the user can versatility exploit existing concepts found within this same PIMO. Therefore, SemChat is integrated with NEPOMUK through its PIMO component where the location, person and organization concepts are used to store the extracted concepts from a chat conversation. A better integration of SemChat with NEPOMUK's other components will be investigated in the future.

We opted to go for a multi-protocol based chat client rather than a single protocol such as Skype or MSN because this includes the possibility to connect to multiple chat protocols from within the same client. Spark IM⁴ was found to be an ideal candidate for SemChat because apart from being open source, it could be further extended through plug-in development.

The extraction mechanism we opted for is based on XtraK4me⁵ key-phrase extractor and ANNIE⁶ named entity recogniser (NER), which is a component within GATE⁷, since we require the extraction of the most important key phrases from a chat conversation and the identification of their entities. The main reason behind utilizing XtraK4Me was based on the fact that it makes use of several

⁴ <http://www.igniterealtime.org/projects/spark/index.jsp>

⁵ <http://smile.deri.ie/projects/keyphrase-extraction>

⁶ <http://gate.ac.uk/ie/annie.html>

⁷ <http://gate.ac.uk/>

GATE components and can also extract key phrases from both text documents and string representations unlike other key phrase extractors which were considered. On the otherhand, ANNIE NER is able to identify multiple entities and can also be extended to recognize user defined entities through JAPE⁸, unlike other NERs considered.

Though various chat clients have a search facility, such as the case of Skype, this is limited in its capabilities. We intend to extend Spark's search facility to go over the extracted content and to allow for interesting searches such as searching by date or concept name to find any semantically related concepts.

2.1 SemChat's Concept Extraction Mechanism

When SemChat is enabled by the user, it monitors chat sessions and upon detecting the closure of a chat session or a chat room within Spark, SemChat starts its main processing. The reason behind the use of the end of chat session as a trigger for SemChat to provide useful information to the user, was mainly motivated by the requirement to implement the system as a non-intrusive one. By adopting this approach, SemChat in fact, strives to limit the cost of interruptions, which as described by [7] varies on average between 10 to 15 minutes before the users returned their focus to the disrupted task, which in this case would be the current chat activity.

SemChat starts by first retrieving the chat conversation between both users and passes this to the XtraK4Me key phrase extractor, which in turn identifies the main key words within a chat instance and finds the ones which are not already stored within the user's PIMO in NEPOMUK, by the use of NEPOMUK's search feature. All unique key phrases are then passed through ANNIE, so that their entities can be identified. ANNIE is able to recognize typical entities such as locations, persons, organizations and dates.

Once this process is complete the user is presented with a notification linked to a list of extracted concepts which is displayed in a separate tab. The intention behind this feature is to make the whole process less disruptive and distracting, as explained earlier. Context menus, as can be seen in Figure 2, are used to allow the user to choose to save a concept within the user's PIMO within NEPOMUK, thus confirming the importance and relevance of this concept, to delete a concept, indicating to SemChat that the concept is not relevant or to retrieve more information about the concept. In case the user chooses the first option, she can then also check that this concept was successfully stored within her PIMO and under the correct category. In case she wants more information about a particular concept, we have used Wikipedia⁹ as our information repository, with snippets of information retrieved being displayed appropriately in a separate pop-up window.

The process of extracting possible events from a chat conversation is slightly different from that described above. In this case the whole chat conversation is

⁸ <http://gate.ac.uk/sale/tao/splitch8.html#x12-2080008>

⁹ <http://en.wikipedia.org/>



Fig. 2. A context menu showing the three options presented for each concept

passed directly through ANNIE to extract any existing events. Since by default ANNIE does not handle such entities it had to be extended. This was done by implementing a number of JAPE rules that specify how to recognize possible events within a chat conversation using regular expressions in annotations, as can be typically seen in Figure 3.

```
Phase: EventAnnotations
Input: Lookup DateClass
Rule: EventRule
(
  { Lookup.majorType==event_trigger }
):eventTrigger
-->
{
  AnnotationSet matchedAnns= (AnnotationSet)bindings.get("eventTrigger");
  FeatureMap newFeatures= Factory.newFeatureMap();
  newFeatures.put("rule","EventRule");
  outputAS.add(matchedAnns.firstNode(),matchedAnns.lastNode(),
  "EventTrigger",newFeatures);
}
```

Fig. 3. JAPE rule for annotating a sequence of text referring to a meeting

The implemented JAPE rules look up for different kinds of text sequences, such as phrases that may indicate a possible meeting, and different types of dates and time. Figure 3 shows the JAPE rule that was implemented to look up phrases which might indicate a possible event within a conversation. The *EventRule* rule will match any text that is an annotation of the *event_trigger* grammar. An *event_trigger* grammar consists of several phrases such as “*Meeting at*” and “*meeting with*” amongst others, which can all indicate a possible meeting. Once this rule matches a sequence of text, the whole sequence is allocated a label by

the rule, in our case this is *eventTrigger*. When this process is complete, any extracted events are also presented to the user in a separate tab.

For each extracted event, the user has the possibility to edit both the title of the event and also the prospective date details as are required. Any annotated event will automatically also be saved within Spark’s Task List event scheduler as depicted in Figure 4. The user will be reminded of any forthcoming events by means of a notification on the event’s due day.

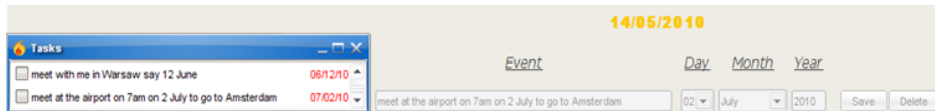


Fig. 4. The saved Event in the Spark’s Task list event scheduler

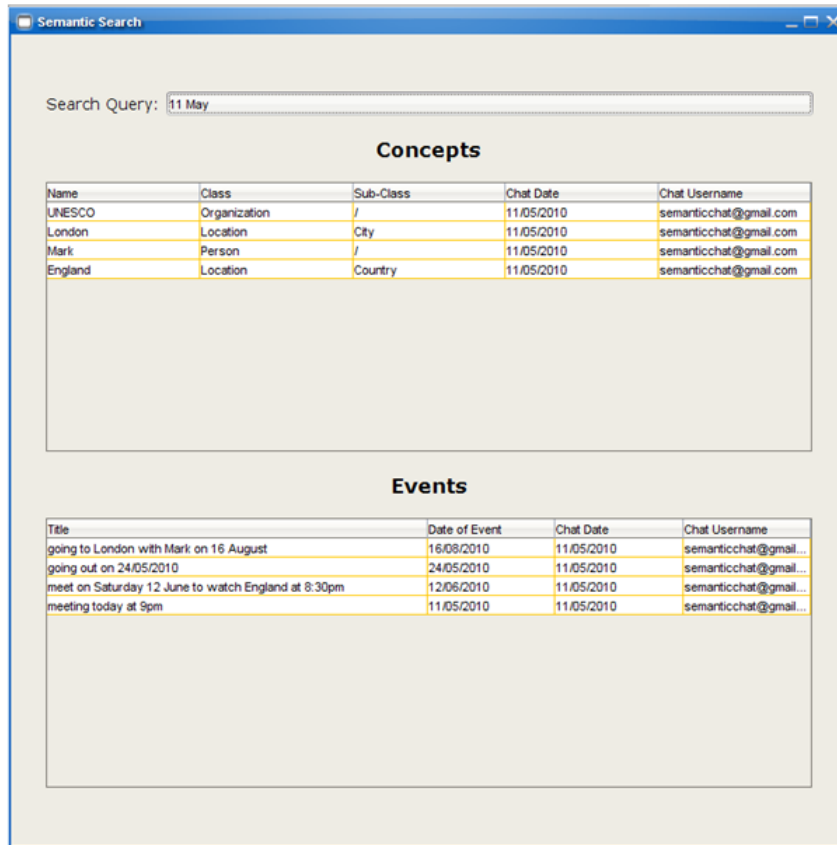
Whenever a user switches off or logs out of Spark, any extracted concepts that were not annotated or deleted, within the current session, will be cached in a RDF storage. A list of such pending concepts is displayed to the user the next time that she enables the SemChat plug-in. We have implemented this feature in this manner so that the user would have “*another chance*” to annotate such concepts if deemed relevant. The concepts that are saved by the user are also cached in a separate RDF storage since they are used by the semantic search feature which will be discussed in the following section 2.2.

It is important to note that any deleted concepts are not cached, and they will be presented again to the user if they are extracted during another chat conversation. The reason behind this implementation is that some concepts, which are not deemed important during a particular chat, could still be seen as important during some future chat which has a different context. For example *during a particular chat session the name of David Guetta is mentioned. However at the time the user did not deem this to be important and deleted the extracted information. Nevertheless, during another chat conversation which was about the Isle of MTV show and which listed the said DJ as one of the participants, the user decided to annotate the person concept and find more information about it.*

2.2 SemChat Search

The semantic search feature helps the user to retrieve any of the annotated concepts. The user can filter-out a search by a number of defined criteria for example by date, whereby she will be returned with any semantically related concepts that satisfy these search criteria. This feature was implemented so that if a user needs to find some previous concepts, such as for example a previously annotated event, she can do so with ease, without the need to go through the whole chat transcripts. Each concept retrieved is presented to the user with its

full details and a typical example of a result obtained from the SemChat's search can be seen in Figure 5.



The screenshot shows a window titled "Semantic Search" with a search query of "11 May". Below the search bar, there are two sections: "Concepts" and "Events".

Concepts

Name	Class	Sub-Class	Chat Date	Chat Username
UNESCO	Organization	/	11/05/2010	semanticchat@gmail.com
London	Location	City	11/05/2010	semanticchat@gmail.com
Mark	Person	/	11/05/2010	semanticchat@gmail.com
England	Location	Country	11/05/2010	semanticchat@gmail.com

Events

Title	Date of Event	Chat Date	Chat Username
going to London with Mark on 16 August	16/08/2010	11/05/2010	semanticchat@gmail...
going out on 24/05/2010	24/05/2010	11/05/2010	semanticchat@gmail...
meet on Saturday 12 June to watch England at 8:30pm	12/06/2010	11/05/2010	semanticchat@gmail...
meeting today at 9pm	11/05/2010	11/05/2010	semanticchat@gmail...

Fig. 5. Semantic search results

3 Evaluation

A usability session was organized as an initial effort to evaluate SemChat. In our setup we considered findings from previous research by [4] which outlined that 6-12 participants are enough to test the usability of a system and provide enough useful information such that initial but concrete conclusions can be made.

In line with this idea 8 participants, mostly students and colleagues, took part in this evaluation exercise. The evaluation session was split into three parts: the first part consisted in the exposure of SemChat's features through a walk-through

example; the second part involved each of the participants getting accustomed to SemChat by chatting with another participant for approximately 20 minutes; the third part consisted in each participant filling in a questionnaire which targeted several aspects of the system.

From the evaluation process, we were able to identify both the limitations as well as possible improvements that we could, in future, affect to our system. Based on the initial results, we could positively conclude that SemChats' main features of extracting concepts (and events) from a chat conversation and that of providing further information through Wikipedia, proved to be popular and useful features amongst the participants. The same can be stated for the integration of SemChat with Spark's event scheduler. It is important to note that the time that the extraction process takes depends on the length of the chat conversation since the more text there is, the more time it takes to extract the whole text. From the evaluation conducted, it was found that it took between 3 to 5 seconds to extract a conversation of approximately 20 minutes.

The semantic search feature was deemed to be less important by 50% of the participants, primarily because they did not find the need to search for any past annotated concepts. This is understandable, since the chat session was rather short. Yet another reason behind this could be attributed to the fact that participants were not accustomed to searching within chat conversations, since the majority of well-known chat clients, provide only limited search facilities, and thus possibly participants were unaware of the potential behind a semantic search facility. On the other hand, there was a high level of satisfaction amongst the other 50% of the participants who used the semantic search facility.

In some cases, however, important concepts flagged within a conversation were not extracted. We attributed this to the fact that the XtraK4Me key phrase extractor selects the most important key phrases according to their occurrence rate. In the future, this problem will be addressed by tweaking XtraK4Me.

It was also noted that in some cases, the event concepts were not being extracted, as expected. This was due to the fact that the events did not conform to the structure that SemChat's events extraction mechanism was implemented to recognize. An example of such an event was "*will be going to Holland*", since no date or person's name was included in the phrase indicating such an event.

A possible solution for this limitation is to further extend ANNIE to recognize other different types of events that could be present within a chat conversation, however this might still not solve the problem completely. In [1] the use of *pidgin languages* is suggested to limit the different ways in which people record information in a note-taking tool, however this could be complicated to learn and at the end could possibly also be counter-productive.

In [2], the main problematic issues related to extracting information from chat are thoroughly analysed. Due to the "*noisy*" nature of chat content, in particular the fact that it may contain misspellings, non-standard use of orthography, punctuation and grammar, presents difficulties for generic information extraction engines. Furthermore the possibility of having "*interleaving of multiple topics and the effects of a dynamic, interactive mode of discourse where semantic*

content changes as the discourse progresses”, makes it even more troublesome. The suggested solution, by [2], is based on a chat-specific, information extraction engine [13], that is capable of performing robustly when faced with such “*surface noise*” by typically allowing for chat data that contains non-standard orthography, punctuation, spelling and grammar.

4 Related Work

In this section we discuss some research which inspired our work on SemChat. The considered research is focused on the extraction of semantic information from notes and chat conversations.

ConChat [9] is a context-aware chat program which improves electronic communication by presenting contextual information. It tries to solve semantic conflicts which occur in chat conversations through the tagging of potentially ambiguous chat messages. ConChat solves part of this problem and therefore is a step forward towards eliminating semantic conflicts which occur in chat sessions. SemChat was designed in a way that it caters for some of the semantic ambiguities related to time, currency, units of measurements and date formats in a similar way to that in ConChat. In the case of time and date formats, this problem is catered in a different manner from ConChat since several JAPE rules were implemented to recognize different types of date formats that can be used within a chat conversation.

GaChat [6] on the other hand uses morphological analysis to extract the proper nouns from the dialogue text. Online images and articles from Wikipedia which are related in a way to these extracted nouns are simultaneously displayed alongside the dialogue text. This additional data is automatically displayed on the chat windows of both user and sender of the message to help reduce the elements of ambiguity like searching and also the asking of some particular details of a particular phrase. In the case of SemChat, the user has the option to seek further information from Wikipedia about each extracted concept.

SAM [5] tries to identify a number of problems that IM systems encounter in order to try to improve the content management of IM systems, moving towards the Networked Semantic Desktop. SAM extends a chat client by semantic annotations, semantic search, semantic browsing and semantic meta-data communication. SAM’s chat window offers a taxonomy panel where the annotation of messages is permitted whilst a user is chatting. SemChat is similar to SAM, however in our case we extend Spark, which is also an XMPP protocol client, with the semantic annotations of concepts extracted from a chat conversation and with a semantic search feature based on the concepts that are annotated by the user. Nevertheless, within SemChat we store extracted concepts within NEPOMUK’s PIMO and events are linked to an event scheduler, making SemChat more versatile and in line with PIM tools.

Though not directly related to semantic chat as the research mentioned above, Semanta[11] which is a plug-in to two popular email clients has some similarities to SemChat which are important to mention. Firstly this system

uses the existing email transport technology and fully integrates with NEPO-MUK. This is similar to SemChat, in fact the architecture behind our semantic chat client was inspired by Semanta. Secondly Semanta handles and keeps track of action items within email messages and also extracts tasks and appointments from email messages which are then added to the email client's scheduler. In a similar fashion, through SemChat it is possible to extract events from chat conversations which are manually annotated by the user and which are stored within Spark's task list scheduler. In this respect SemChat tags along the approach adopted by Semanta and not merely adds a semantic component over the traditional chat component, as was mainly done in the research mentioned above, but strives to become a PIM tool in all respect.

5 Future Work

With regards to future work we have a number of interesting ideas, including the integration of SemChat with popular applications such as a popular email client like Thunderbird¹⁰. Through this integration any extracted events could be logged automatically into the email client's event scheduler, rather than keeping this information only available to Spark's task list event scheduler.

As already mentioned in Section 3, it is envisaged that other types of entities could be extracted from chat conversations, apart from the ones already identified. Typical examples of such entities could be, emails, products, addresses and telephone numbers. In this case, ANNIE would need to be further extended through JAPE in a similar manner adopted for events. The solution based on dedicated JAPE rules might however not always turn up each and every existing entity within a chat, due to the fact that chat data is inherently noisy, as explained in [2]. We are nevertheless confident that this approach complimented by user feedback can still achieve a satisfactory level of precision in identifying those concepts which are relevant for the user's PIM.

The semantic search feature could also be improved in several aspects. One such aspect is to further optimize the searching process since it has to sift through many annotated concepts and it takes some time to find all the semantic relations between the concepts satisfying the search criteria. The inclusion of an auto-completion facility, would make it easier for the user to retrieve the semantically related concepts in a faster and more efficient way.

This search facility could also be further enhanced such that it would display the part of the chat transcript from where each concept satisfying the search criteria was retrieved. Through this enhancement the user would be able to better recall the context within which a particular concept was mentioned during a chat conversation.

The semantic annotations generated by SemChat could also be quantitatively evaluated in the future. In this case the users could be assigned a set of tasks that will be conducted initially on a normal chat client and then performed also

¹⁰ <http://www.mozillamessaging.com/en-US/thunderbird/>

on SemChat. This form of analysis might provide us with further insights into the costs and benefits of using such a semantic chat client for predefined tasks.

6 Conclusion

In this paper we presented SemChat, which is our initial effort at integrating a semantic chat component with a social semantic desktop, NEPOMUK. With the area of PIM increasingly becoming important, SemChat contributes further to this area through the integration of concepts in the user's PIMO as well as the integration of events with an events scheduler. Although the initial evaluation of the developed prototype is very encouraging, further work is required so that SemChat evolves into a fully realised PIM tool.

References

1. Michael Bernstein, Max Van Kleek, Mc Schraefel, and David R. Karger. : Evolution and Evaluation of an Information Scrap Manager. In CHI 2008 Workshop on Personal Information Management, Florence, Italy (2008)
2. Cassandre Creswell, Nicholas Schwartzmyer, Rohini Srihari: Information extraction for multi-participant, task-oriented, synchronous, computer-mediated communication: a corpus study of chat data. In Proc. IJCAI-2007 Workshop on Analytics for Noisy and Unstructured Text Data, Hyderabad, India, pp. 131-138 (2007)
3. Laura Dragan, Siegfried Handschuh. : SemNotes- Note-taking on the Semantic Desktop. In poster session of the 6th European Semantic Web Conference, ESWC'09, Heraklion, Crete, Greece (2009)
4. Joseph S. Dumas, Janice C. Redish. : A Practival Guide to Usability Testing (Revised Edition). Intellect Books, Exeter, UK (1999)
5. Thomas Franz, Steffen Staab. : SAM: Semantics Aware Messenger for the Networked Semantic Desktop. Koblenz-Landau, Germany (2008)
6. Satoshi Horiguchi, Akifumi Inoue, Tohru Hoshi, Kenichi Okada. : GaChat:A chat system that displays online retrieval information in dialogue text. In Workshop on Visual Interfaces to the Social and the Semantic Web(VISSW2009), Sanibel Island, Florida (2009)
7. Shamsi T. Iqbal, Eric Horvitz. : Disruption and Recovery of Computing Tasks: Field Study, Analysis, and Directions. In CHI '07: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, San Jose, California, USA, pp. 677-686 (2007)
8. NEPOMUK How To. NEPOMUK Social Semantic Desktop. <http://dev.nepomuk.semanticdesktop.org/wiki/UsingNepomuk> (2008)
9. Anand Ranganathan, Roy H. Campbell, Arath Ravi, Anupama Mahajan. : Con-Chat: A Context-Aware Chat Program. IEEE Persuasive Computing, Vol. 1, Issue 3 (2002)
10. Simon Scerri, Brian Davis, Siegfried Handschuh, Manfred Hauswirth. : Semanta - Semantic Email made easy. In Proceedings of the 6th European Semantic Web Conference, ESWC'09, Heraklion, Crete, Greece, pp 36-50 (2009)
11. Simon Scerri, Ioana Giurgiu, Brian Davis, Siegfried Handschuh. : Semanta - Semantic Email in Action. In Proceedings of the 6th European Semantic Web Conference, ESWC'09, Heraklion, Crete, Greece, pp 883-887 (2009)

12. Michael Sintek, Siegfried Handschuh, Simon Scerri, Ludger van Elst. : Technologies for the Social Semantic Desktop. In Reasoning Web. Semantic Technologies for Information Systems: 5th International Summer School 2009, Brixen-Bressanone, Italy (2009)
13. Rohini K. Srihari, Wei Li, Cheng Nium, Thomas Cornell. : InfoXtract: A Customizable Intermediate Level Information Extraction Engine. In Journal of Natural Language Engineering, Cambridge U. Press, 14(1), pp.33-69 (2008)

Ad-hoc File Sharing Using Linked Data Technologies

Niko Popitsch and Bernhard Schandl

University of Vienna, Department of Distributed and Multimedia Systems
{niko.popitsch|bernhard.schandl}@univie.ac.at

Abstract. A large fraction of our information, both in the professional and private domains, is stored in the form of files on our personal computers. When we collaborate with co-workers or meet with friends, mechanisms for sharing files and file annotations are frequently required. However, centralized file sharing infrastructures are often not available or complicated to set up, and approaches like peer-to-peer sharing infrequently provide functionality beyond simple copying of files between machines. In this paper we present a light-weight approach for ad-hoc file sharing based on Linked Data principles. Our system exposes parts of a file system as Linked Data and allows users to interlink and annotate resources in such linked file systems. We further provide a mechanism for mounting multiple such file systems together, and for seamlessly navigating them using a common Web browser. As the exposed files and directories become Web resources, they are amenable to a large set of Semantic Web and Linked Data tools. Human and machine users may exploit such linked file systems in ad-hoc data sharing scenarios. They may further add arbitrary annotations to local and remote linked file system resources, which may also be shared among users. Finally, file system objects may be searched based on their extracted metadata and such semantic annotations.

1 Introduction

File sharing has become a central activity in the professional and private domains [1–3]. The sharing of files is supported by a large number of tools and methods, ranging from email attachments over centralized file servers to peer-to-peer sharing applications. An increasingly used method is the exchange of data via Internet-based sharing systems that may be specialized for a certain media type (e.g., Flickr, Youtube) or of general purpose (e.g., DropBox). Users select one or multiple of these tools to solve a particular sharing problem based on *what* is shared and with *whom* it is shared [1].

In this work we focus on a particular type of data sharing, *ad-hoc sharing*, which is characterized by the lack of pre-existing sharing infrastructure. Often, the participating users and their devices are physically close; however, this is not a precondition. Ad-hoc sharing is rather identified by the need to quickly exchange data with users or devices they do not often share data with (in the

past and the future) so that the setup of heavyweight sharing infrastructures is unfeasible [2].

Consider for example the following scenario: after a common vacation, Alice and Bob meet with friends to talk about their common experiences and exchange their digital photos among each other. Alice would like to give their friends a photo presentation of her and Bob’s photos. Bob would like to copy some of Alice’s photos to his machine but first needs to select which ones. Further, Bob would like to add information about where a particular photo was taken (e.g., what restaurant they had that great fish menu at). These annotations should be accessible also to Alice and their friends, and they should be able to extend them. Bob would further like to explicitly link related photos, e.g., he would like to relate the photos of his daughter taken during last year’s vacation to this year’s photos.

Although Alice, Bob, and their friends know each other well, it is rather unlikely that they exchange data frequently; therefore the introduction of a heavyweight sharing infrastructure might be immoderate. Note that this scenario does not require the actors to meet in person—everything could also be done remotely. The scenario mentioned before could partly be solved with a centralized, online data sharing application. This would, however, raise the following issues:

1. All involved devices would require Internet access, although local connectivity would be sufficient for most tasks.
2. The annotation tasks would be restricted to the functionality offered by the particular application.
3. By uploading data to an online platform, digital copies of these data are created. Annotations would refer to these copies rather than to the original files. When a user decides to manipulate a data item locally (e.g., applying a photo filter to reduce the red-eye effect), they would need to update the data manually on the online platform so that others can access this improved version.
4. Storing data on Web servers usually raises security and privacy issues.
5. Many existing sharing platforms handle only particular file types.

In this paper we present an alternative method for ad-hoc sharing based on Linked Data. We present how our filesystem, TripFS [4], can be used to expose parts of a local file system as Linked Data, and how multiple such *linked file systems* can be mounted and seamlessly navigated with a common Web browser. As the exposed files and directories become regular Web resources, they are amenable to a large set of Semantic Web and Linked Data tools. We further describe how arbitrary annotations and links can be added to such resources: resources may be linked to local and remote files exposed via TripFS, but also to any other Web resource or Linked Data source. We describe how human and machine users may exploit such linked file systems in ad-hoc data sharing scenarios as the one presented above, and conclude with a discussion of advantages and shortcomings of our approach when compared with related work from the file sharing domain.

```

1 <http://queens.mminf.univie.ac.at:9876/resource/71023c2f-8aec-41b0-ac0b-0ce38cf1e0f7>
2   a tripsfs:File ;
3   rdfs:label "piran2.jpg" ;
4   tripsfs:local-name "piran2.jpg" ;
5   tripsfs:path "file:/g:/watch/images/2009/vacation/piran/piran2.jpg" ;
6   tripsfs:size "46170"^^xsd:long ;
7   tripsfs:modified "2010-07-20T10:04:59"^^xsd:dateTime ;
8   tripsfs:parent
9     <http://queens.mminf.univie.ac.at:9876/resource/3bb652a5-d38c-4c01-b9b7-548c0c19e546> ;
10  nfo:hasHash "58717"^^xsd:int ;
11  nie:mimeType "image/jpg" ;

```

Fig. 1. RDF representation of a file served by TripFS. In addition to basic file system data (lines 1–7), the representation contains a triple that connects the file to its parent directory (lines 8–9) and extracted metadata (lines 10–11).

2 TripFS: Exposing File Systems as Linked Data

TripFS¹ [4] is a lightweight utility that publishes parts of a local file system as Linked Data. It bridges the gap between the distinct worlds of hierarchical file systems and the hyperlink-based Web by

1. providing *stable, de-referencable URIs* for directories and files, thereby making it possible to establish stable references to local and remote file system objects;
2. *extracting metadata* from files, thereby allowing to find and access files based on their contents instead of their location;
3. *linking files* to external Linked Data sources based on extracted metadata, thereby opening file systems for global, enterprise-wide, or personal information integration; and
4. *servicing* file and directory descriptions as Linked Data (through de-referencable URIs, a SPARQL endpoint, and RDF representations), thereby providing access to file systems using standardized (Semantic) Web technologies.

TripFS combines several third-party components (including the Jena Semantic Web Framework², Aperture³ for metadata extraction, the Jetty HTTP Server⁴, and the DSNotify monitoring framework [5]) and can be deployed as a background process on any Java-enabled computer. It can be configured to use any RDF storage backend for storing annotations and extracted metadata. Upon start, it crawls the configured file system subtrees and builds an RDF representation where directories and files are represented as RDF resources. TripFS extracts metadata from file system objects and links these objects with each other, as well as with external data sources. After crawling, DSNotify is used to monitor changes in the file system, which are in consequence reflected in the

¹ <http://purl.org/tripfs>

² <http://openjena.org>

³ <http://aperture.sourceforge.net>

⁴ <http://jetty.codehaus.org>

RDF model. New or changed files are re-analyzed, so that the RDF model remains in sync with the local file system. Figure 1 shows an RDF description of a file, as served by TripFS. In addition to the RDF representation, TripFS provides a convenient HTML-based interface that allows the user to navigate through the file hierarchy. All main components of TripFS are flexible and extensible; in particular, extractors (e.g., for new file types) and linker components (for arbitrary external data sources) can be added easily.

3 Linked Data-style Ad-hoc File Sharing

In *ad-hoc file sharing*, users that do not exchange data regularly (in the past and in the future) have the short-term need to exchange file-based contents between multiple machines. As discussed, they cannot resort to permanent infrastructure (like file servers, hosting providers, or Web applications) as it is either unfeasible to set-up such an infrastructure or due to infrastructural constraints (e.g., limited connectivity, firewalls, etc.). Often, ad-hoc file sharing takes place in situations where users are co-located and have some but limited shared network infrastructure (e.g., a Wi-Fi network). Ad-hoc file sharing is of relevance both in professional and in private contexts: for instance, during a business meeting one may want to share a certain document or spreadsheet with all participants. In the private domain, one may want to exchange photos from the recent vacation with friends during a relaxed dinner.

Analyzing the related works from the file sharing domain mentioned in this paper (in particular, [2] and [6]) and combining it with our own considerations led us to the following list of requirements for ad-hoc file sharing:

1. *Universality*: all file types should be sharable.
2. *Minimum preconditions*: participants (i.e., data providers and consumers) should not require a lot of additional software to be able to share files.
3. *Minimum configuration*: setting-up a new collaborative file space should be as easy as possible.
4. *Lightweight and usable access control*: it should be simple and fast to assess shared files and to decide on access rights.
5. *Platform and network independence*: it should be possible to share files across different hardware and software (operating system) platforms. It should further be possible to share files across network boundaries.
6. *Support for transient data and stable links*: data in ad-hoc sharing scenarios is accessible only for a short amount of time. Sometimes this is sufficient in a particular sharing scenario [2]. However, sometimes operations on shared data run over multiple such ad-hoc sessions (in our case, e.g., annotations and links between files should be preserved) and sharing solutions should support such operations.

3.1 Ad-hoc File Sharing in Practice

Today, sending email attachments seems to be the predominant way of personal file sharing [3]. Volda et al. analyzed that users tend to fall-back to such a uni-

versal data sharing mechanism when they are unsure about the availability of a certain sharing tool at the recipients side, or when they have problems of communicating through firewalls [1]. Another common technique for infrastructure-less ad-hoc file sharing is to use detachable physical devices, like USB sticks. Usually, this “offline” method for file sharing works straightforward, except for limited storage capacity on the removable media. Another popular way to share files is to send them via instant messaging (IM) channels. Most of these tools provide simple mechanisms to send files to one or many chat partners, which however requires all participants to have network connectivity, an account for the IM network, and corresponding client software at hand. This method is further not applicable when the available network does not permit the usage of IM software due to security restrictions, e.g., in corporate intranets. Peer-to-peer based file sharing constitutes another often-used method [1, 3]; however, classical peer-to-peer platforms like Napster, Gnutella or KaZaA seem less applicable in ad-hoc file sharing scenarios.

Other common methods to share files make direct use of the World Wide Web, arguably one of the most important information channels today. The Web is well-supported by most modern devices: even low-capacity mobile devices allow users to access Web resources. It is easy nowadays to set-up personal Web presences without knowing the technical details of content markup and Web hosting. Because of their widespread adoption, Web technologies are a promising candidate for ad-hoc information sharing. However, current Web 2.0 applications that support file sharing suffer from the previously mentioned issues (cf. Section 1) such as the requirement for Internet access or limited annotation support. A major drawback of such centralized systems is that they require all shared files to be uploaded to their Web servers first. In our scenario, this means that Alice would have to upload all her vacation photos before Bob can select some for downloading them to his laptop. These digital objects are not directly connected with their digital “originals” residing on Alice’s computer, meaning that changes to these files are not automatically propagated to the shared versions and vice versa. Further, Alice cannot directly benefit from annotations made to these online copies outside the hosting Web application itself.

In this paper we present an alternative file-sharing approach that does not require a centralized infrastructure or digital copies of resources and is based on Linked Data principles. Linked Data [7] re-uses and extends the Web infrastructure with technologies that allow to represent, transport, and access raw data over the network. In comparison to the traditional, document-centric Web it comprises the significant improvement that it associates resource identifiers (URIs) with structured descriptions that are represented in a unified format (RDF) and can be accessed by de-referencing their URIs. In the context of file systems, Linked Data techniques can be used to expose structured metadata descriptions about files, which allows clients to access them based on their semantic meaning rather than just based on their location in a file system hierarchy [4].

3.2 File Sharing with TripFS

Based on the scenario outlined in Section 1, we have extended TripFS with features that allow users to easily share files across a (local) network, and to connect multiple file systems using Linked Data technologies. In the following we reconsider the scenario and describe which particular TripFS features support this use case.

One-click Sharing. When Alice and Bob meet to discuss and exchange their recent photos, both want to share folders (including subfolders) on their laptops that contain these photos⁵. When Alice starts TripFS on her laptop, it announces its service URL via a Zeroconf⁶ service, so that it can be discovered by other machines on the same network. In parallel, TripFS crawls the selected part of the local file system, extracts metadata from files, and links them to other data sources (cf. Section 2). The resulting triples are incrementally stored in the RDF store and are immediately published via the Linked Data interface.

For adding new directory subtrees to TripFS, Alice makes use of the TripFS Windows Explorer shell extension⁷ that allows to share a folder with a single mouse click (cf. Figure 2). When Alice clicks this button, a local Windows socket is opened and the selected directory's path is sent to TripFS. TripFS adds this directory to its list of exposed root directories and creates a new observed region for DSNotify. The shell extension reports the successful or unsuccessful outcome of this operation to the user via a popup dialog. Immediately, the folder is accessible via the Web server built into TripFS and can be accessed by devices on the network. If the newly exposed root directory lies within a directory that is already exposed, TripFS marks it as inactive in order to avoid unnecessary monitoring and crawling costs for overlapping regions. For the same reasons, TripFS deactivates all existing root directories that lie in a subtree of a newly added root directory.

Accessing Shared File Systems. Since TripFS provides both, an HTML- and an RDF-based view on shared folders, Alice's friends can access her photos using the Web browser installed on their laptops. If their system supports service discovery via Zeroconf they not even have to enter the hostname or IP address of Alice's laptop. They can navigate through the file hierarchy and download their favorite photos (a screenshot of this interface is presented in Figure 3). They could also use the structured data exposed by TripFS to search for files using a visual Linked Data query builder (like, e.g., *Explorator* [8]), which allows to visually construct structured queries. For example, Bob may decide to download only photos taken on a certain day (indicated by EXIF metadata extracted from the photos), or photos that are related to a particular place (represented by

⁵ Let us assume they have access to a shared wireless network.

⁶ Zero Configuration Networking (Zeroconf): <http://www.zeroconf.org>

⁷ It is also possible to add shared directory subtrees via the Web interface.

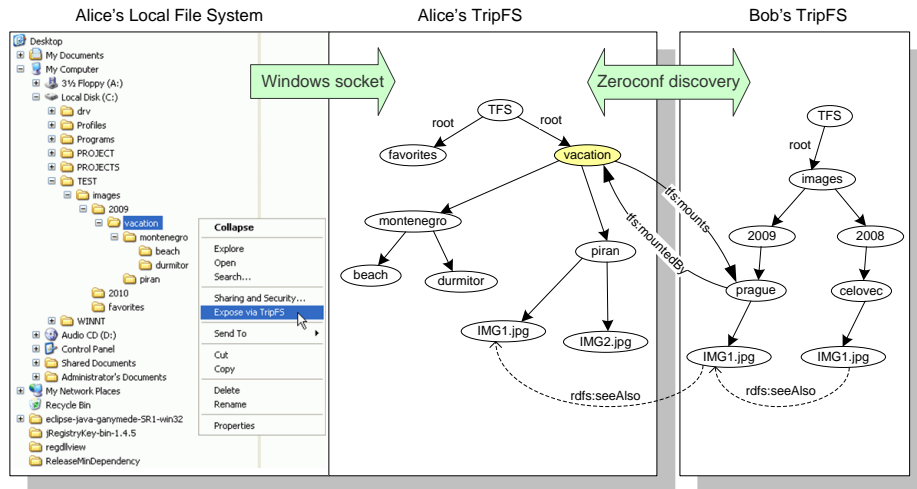


Fig. 2. Ad-hoc file sharing with TripFS: Parts of Alice’s file system depicted on the left have been exposed by her TripFS instance, depicted in the middle. The screenshot on the left shows the Windows explorer shell extension for one-click sharing of file resources via TripFS. The “vacation” resource is the mount point of Bob’s (remote) TripFS. The dashed arrows denote explicit links between files.

a link to a GeoNames entity that has been created based on extracted GPS coordinates).

Annotating Shared Files. While he browses Alice’s photos, Bob wants to annotate one of the pictures because he remembers the particular restaurant where the picture was taken. For this purpose, TripFS provides an *RDF sink*. This component establishes a Web resource that accepts RDF data (for instance, annotations of shared files) sent by clients via HTTP POST, and stores posted triples in a designated named graph within the TripFS RDF store. Later, these annotations are published together with metadata that have been extracted from files. For instance, Bob could drag the URL of the restaurant’s Web page from his Web browser to a designated area on the TripFS HTML interface, causing an `rdfs:seeAlso` triple to be stored. If Bob did this for multiple files, he could later retrieve all photos linked to the restaurant’s Web page through a structured query, as described before.

Mounting Other TripFS Instances. A special form of file annotation is *Linked File System Mounting*. This technique uses Linked Data principles to connect distributed file systems, similar to the well-known *mount* operation in UNIX-like operating systems (cf. Figure 2). TripFS defines an RDF property

`tripfs:mounts`⁸ to link a directory in one instance to a directory in the same or another one. TripFS provides a simple user interface for mounting remote TripFS instances, which leads to the creation of the respective triples in both involved TripFS RDF stores. Applications may add mount links by simply posting a respective triple to the RDF sink. Mount triples should be interpreted by TripFS consumers (such as the Web-based TripFS file browser) like parent-child relationships to enable seamless navigation across file system boundaries. Note that in contrast to local file systems, it is possible to create circular structures using Linked File System Mounting. Although the TripFS RDF sink rejects mount triples that would directly lead to such a situation, circles cannot be generally avoided due to the distributed, open-world nature of Linked Data. Consumers (e.g., crawlers or user interfaces) need to be aware of this possibility to avoid unwished complications like infinite loops. As mount triples reside in the annotation model of the TripFS instance they were posted to, a mount link is initially visible only to clients of this particular instance, as it is the case with UNIX mounts. However, following the idea of the Web of Data, it is reasonable to propagate the mount triple also to the remote TripFS instance, so that it can be easily followed backwards. Thus the RDF sink posts a respective `tripfs:mountedBy` triple to the RDF sink of the remote TripFS. Since TripFS provides stable URIs for files due to its file-monitoring component, these mount points remain valid even if a mounted file system is temporarily unavailable, or if the user decides to move a shared directory to a different location on their hard disk.

Seamless File Browsing. While TripFS allows Alice and Bob to interlink their file systems and mutually add annotations to exposed files, this environment still provides no seamless browsing experience: for instance, file annotations are exposed only by the TripFS instance they are stored in. However, if Bob wants to add a private annotation to one of Alice’s files, it should not be stored in Alice’s TripFS instance but in Bob’s, and he wants this private annotation to appear when he browses Alice’s file system.

To overcome this issue, TripFS contains a Web-based proxy browser that dynamically fetches RDF descriptions from remote sources and enriches them with annotations from the local TripFS RDF store (cf. Figure 3). Annotations are stored in a separate RDF graph in TripFS that is merged with a resource’s (remote or local) RDF graph for rendering purposes. Thus, annotations from the local store that refer (via their subject URI) to resources in the remote source are automatically mashed with the remote source’s RDF descriptions: the user is presented with a single, comprehensive view of remote and local resources.

Duplicate Detection. TripFS provides a simple solution for the detection of duplicate files across multiple file systems. For each published file, TripFS calcu-

⁸ `tripfs:mounts` is a sub-property of the `tripfs:child` property, an inverse property `tripfs:mountedBy` is available. The current version of the TripFS vocabulary is available at <http://purl.org/tripfs/2010/06>.

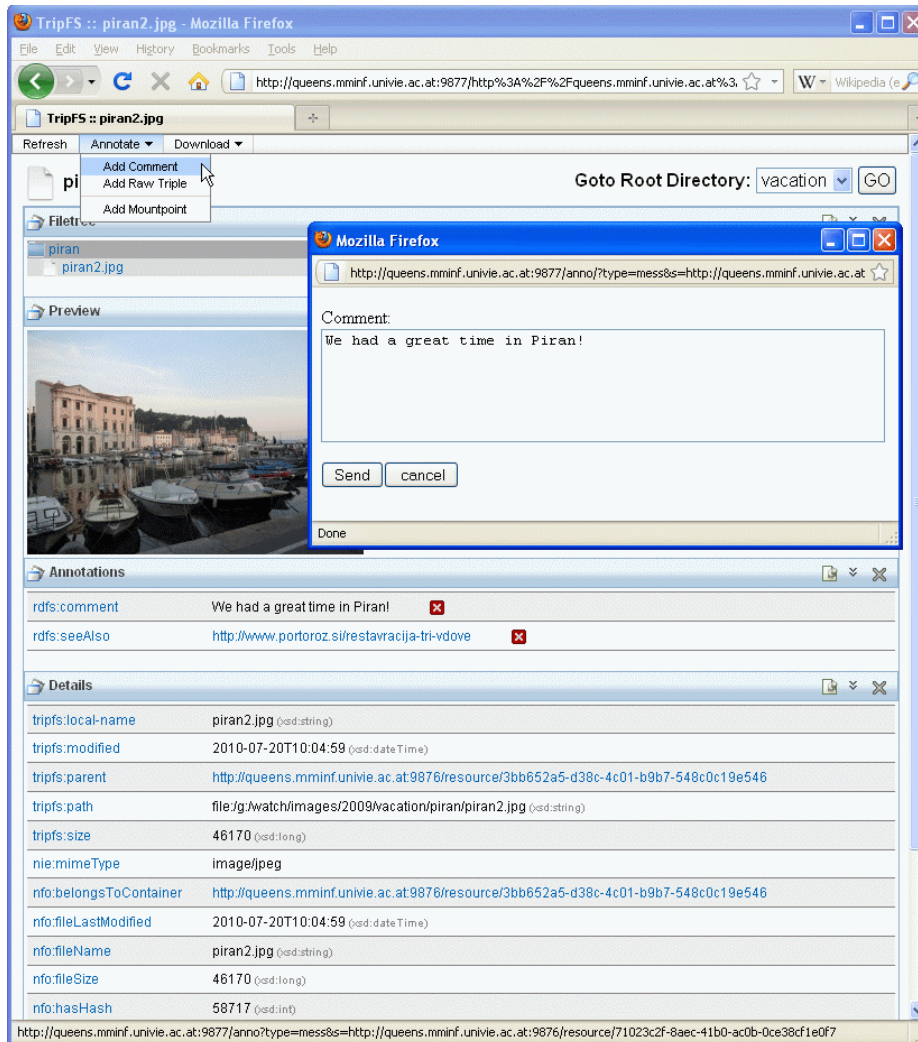


Fig. 3. The Web-based TripFS file browser. This locally running Web application can retrieve local and remote TripFS descriptions and renders them together with annotations retrieved from a local RDF model. Annotations can be added by posting RDF graphs to a servlet or via the Web interface.

lates a content-based checksum and publishes it as property of the file resource. A linker component creates `owl:sameAs` links between files within the published file system, as well as files in other TripFS instances that share a common checksum. For example, when Alice's TripFS is discovered by Bob's TripFS (and vice versa), this linker component is activated and creates `owl:sameAs` links between all duplicates found in Alice's and in Bob's shared folders. By this, Bob is enabled to immediately detect that he already copied a certain file from Alice's laptop last time they met. Further, these `owl:sameAs` links can be exploited to access resource copies when the originals are currently not accessible.

Discovery. Currently, TripFS makes use of a Zeroconf service to discover other TripFS instances. When a new instance is discovered, the duplicate detection linker described above is activated and files with equal checksums are interlinked. One drawback of the current solution is that it is restricted to the local subnet. An alternative method would be to use URNs (e.g., PURLs) for locating physical TripFS addresses. Once the PURL of a particular TripFS instance is known (e.g., because it has been communicated via email), it would remain stable. Disadvantages would be that access to the URN service would be required, and that users have to notify these services whenever their physical address changes (e.g., due to a newly assigned IP address). However, this last step can be easily automated. Another possibility is that the creation of a guest account for a particular TripFS (see below) results in an email that sends an appropriate link to a set of recipients. This link would contain the respective TripFS location as well as the required user credentials for accessing it.

Access Control. The willingness to share data with others often depends on whom these data will be shared with [9]. Access control mechanisms are therefore required also in ad-hoc sharing scenarios. As TripFS is still in a prototypical phase and as security was not our primary research goal, we have no yet implemented access control mechanisms. However, TripFS provides an increased level of privacy and security compared to other sharing platforms since the data remains under full control of the user and is not replicated to external servers.

We are however aware that *usable access control* mechanisms are essential for a system like ours. A first, straightforward solution would be to expose files via HTTPS and introduce password protection, which can be based on the underlying operating system's authentication and permission system. TripFS therefore could reuse already existing mechanisms and would avoid the need to maintain parallel structures. Additionally, a TripFS instance owner could create a new *guest* account that would be valid for a limited time with a single mouse click. The respective credentials could then be transferred to the TripFS consumer via out-of-band methods (e.g., via email or phone). Although this might be sufficient in the discussed ad-hoc sharing scenarios, more fine-grained access control mechanisms and access rights, as discussed for example in [9] and [1], have to be considered in the future.

4 Related Work

Several studies on personal file sharing focused on particular file types (e.g., music or photographs [10, 11]) or on collaboration in corporate intranets.

In [1], the authors analyze several tools and methods for data sharing and report on dimensions for characterizing them. For example, they distinguish between push- and pull-oriented systems and present a user interface for their own peer-to-peer file sharing infrastructure. In accordance with the terminology of that paper, TripFS would be a pull-oriented system that supports public or selective addressing (when password protected) and supports notifications via the DSNotify event log mechanisms [5]. The location of the files during sharing remains the provider's machine.

In [6], Rode et al. identify four significant requirements for their own ad-hoc peer-to-peer file sharing software: (i) zero-configuration for setting up a collaborative file space, (ii) no prior registration of participants required, (iii) no restriction to a fixed infrastructure (e.g., Internet access) and (iv) platform independence. We believe that TripFS meets all these requirements, although the TripFS software has to be installed on all machines that expose their files.

In [2], Dalal et al. identify a number of key problems that are not properly addressed by current data sharing technologies. The authors describe the requirement for *ad-hoc guessting*, where users require transitory, lightweight solutions for sharing data securely with unplanned sets of people with whom they have not previously shared data and that can possibly not be addressed by traditional access control. Similar to Rode et al., they identify minimal setup effort and no need for *a priori* preparations by the participants as key requirements for ad-hoc sharing. Additionally, they encourage the use of universal identifiers (e.g., email addresses) for the identification of users.

5 Conclusions

In this paper we described the current state of TripFS and its extensions since our last publication [4], namely: one-click sharing support; mounting support; seamless file browsing support across distributed, mounted linked file systems; annotation of file system objects and duplicate detection.

We further presented how this *linked file system* can be used in ad-hoc file sharing scenarios. Matching our system to the requirements described in Section 3 we can state that TripFS can be used as a universal file sharing tool that is not restricted to particular file types. TripFS requires no *a-priori* preparations for recipients of shared data. Users that actively share the data need a local TripFS instance that can either be started automatically by the operating system or by double-clicking a JAR file. TripFS allows remote users to browse the shared contents directly on the remote machine before downloading (subsets of) it. It thereby comprises a pull-oriented sharing strategy [1] that is not based on centralized infrastructure like current Web 2.0 applications. Sharing a directory subtree with TripFS is made easy by its Windows shell extension, and we consider the development of comparable tools for other operating systems.

TripFS has been implemented in Java and can be used on all platforms that support Java 1.6 or higher. In the future we aim to explore how TripFS can be deployed on mobile devices like cell phones, which are presumably more often involved in ad-hoc sharing situations. Then, TripFS could additionally be useful in “*sharing with myself*” scenarios [2], e.g., for copying photos from a person’s cell phone to a desktop computer or vice versa.

References

1. Stephen Volda, W. Keith Edwards, Mark W. Newman, Rebecca E. Grinter, and Nicolas Ducheneaut. Share and Share Alike: Exploring the User Interface Affordances of File Sharing. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 221–230, New York, NY, USA, 2006. ACM.
2. Brinda Dalal, Les Nelson, Diana Smetters, Nathaniel Good, and Ame Elliot. Ad-hoc Guesting: When Exceptions are the Rule. In *UPSEC'08: Proceedings of the 1st Conference on Usability, Psychology, and Security*, pages 1–5, Berkeley, CA, USA, 2008. USENIX Association.
3. Tara Whalen, Elaine Toms, and James Blustein. File Sharing and Group Information Management. In *Personal Information Management: PIM 2008*, 2008.
4. Bernhard Schandl and Niko Popitsch. Lifting File Systems into the Linked Data Cloud with TripFS. In *3rd International Workshop on Linked Data on the Web (LDOW2010) - Raleigh, North Carolina, USA*, 2010.
5. Niko Popitsch and Bernhard Haslhofer. DSNotify: Handling Broken Links in the Web of Data. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 761–770, New York, NY, USA, 2010. ACM.
6. Jennifer Rode, Carolina Johansson, Paul DiGioia, Roberto Silva Filho, Kari Nies, David H. Nguyen, Jie Ren, Paul Dourish, and David Redmiles. Seeing Further: Extending Visualization as a Basis for Usable Security. In *SOUPS '06: Proceedings of the second symposium on Usable privacy and security*, pages 145–155, New York, NY, USA, 2006. ACM.
7. Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data — The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3), 2009.
8. Samur Araujo and Daniel Schwabe. Explorator: A Tool for Exploring RDF Data Through Direct Manipulation. In *Proceedings of the 2nd International Workshop on Linked Data on the Web (LDOW), Madrid, Spain*, 2009.
9. Judith S. Olson, Jonathan Grudin, and Eric Horvitz. A Study of Preferences for Sharing and Privacy. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1985–1988, New York, NY, USA, 2005. ACM.
10. Barry Brown, Abigail J. Sellen, and Erik Geelhoed. Music Sharing as a Computer Supported Collaborative Application. In *ECSCW'01: Proceedings of the seventh conference on European Conference on Computer Supported Cooperative Work*, pages 179–198, Norwell, MA, USA, 2001. Kluwer Academic Publishers.
11. Andrew D. Miller and W. Keith Edwards. Give and Take: A Study of Consumer Photo-sharing Culture and Practice. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 347–356, New York, NY, USA, 2007. ACM.

Towards a Simple Textual Trace Based Personal Exo-Memory

Pierre Deransart

INRIA Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France
Pierre.Deransart@inria.fr

Abstract. This paper presents an experimentation with a continuously updated textual exo-memory used to assist the natural memory of a subject. It shows how trace theory could be used to improve the device. Main characteristics of such a memory aid are maintenance by the subject, limited size, plasticity, and persistence of the recall quality in the long term.

1 Introduction

This paper is an attempt to define a digital artifact capable of serving as an personal exo-memory¹, with the aim to strengthen the biological memory. It does not attempt to address proved memory deficits nor to treat diseases associated with aging, although this study could help to design adapted artifacts. On the contrary, it is assumed that the user has no brain deficit, and that he wants to improve his practical skills of life. This requires having access at any time to some detailed information (of a person, a visited venue, a past event, a current affair, a personal curiosity ...), which may help to recall facts from biological memory.

A simple example is a current telephone alphabetically organized notebook (paper or digital), the names of which are associated with information such as phone number, address, and other details. Another common support is a notebook for working notes or of personal diary which includes series of paragraphs describing daily events. If the support of the book is physical such as paper, the difficulty of retrieving information increases in relation with the total size. In practice therefore, finding a phone number in such a book can become impossible or very cumbersome if the name of the person to call has been forgotten. A notebook may become useless because very related events may become widely separated in the medium (for example reports from periodic meetings on a given topic) and recall of some decision may become virtually impossible.

¹ or external memory. This artifact falls into the category HDM (Human Digital Memories) and has links with the areas PIA (Personal Information Archive), or PIM (Personal Information Management).

Today there is a huge number of digital media offering not only the basic functions mentioned above, but also a great variety of services such as collective or shared calendar, handwritten computer aided notes which are immediately stored. Moreover, the exponential growth of digital storage media, beyond the simple Moore's law² suggests that almost all events of life can be stored on a single hard drive. In [2] the authors estimate that around 16GB per year are sufficient to store all elements of the daily life of a person (including all elements of life context, emails, sounds, images, videos and music). This offers today the possibility to keep on a digital medium a very detailed trace of all activities and social interactions performed during the whole life of a single individual. Even if the speedup of the means of communication and exchange suggests that these volumes can be underestimated, it is now possible to consider retaining all its "memories" on a single private digital medium, with outstanding opportunities to navigate in this ocean of memories.

It must first be noted that outsourcing one's own memory in order to assist it or just to keep it, takes up a large amount of human activity. Writing notes, categorizing them, using a calendar, building a personal library or documentation, organizing its environment, all contribute in achieving such a goal. Elderly people live in an environment full of objects "memory"³ which contribute to their quality of life, i.e. helping them to preserve some of their memory. If the idea of auxiliary memory may be extended to include large social systems, this study is restricted to what is usually called "personal memory", closer to a sophisticated booknote aimed at accompanying the person at every moment of his/her life.

This article includes four main sections. In the first (Section 2) we characterize what we mean by exo-memory. The second (Section 3) describes an experiment with an exo-memory represented by a text file handled with a text editor. The third (Section 4) gives some possible theoretical foundations using the notion of trace, and the last one (Section 5) discusses some features that are essential to this type of exo-memory and some limitations. A full version can be found on archive [4].

2 Personal Exo-memory

In this section we identify the essential characteristics of what we call here exo-memory.

In 1945 Vannevar Bush wrote an article entitled "As We May Think" in which he laid the foundation of Memex [5]: "a device in which an individual can

² A version of the law known as "Moore's law" states that computer processing power has doubled every two years since 1969. This exponential growth is three to four times higher for digital storage media [1].

³ Collection of objects likely with a strong emotional charge which correspond to meaningful life facts and help to recall them. This is very well described in the romance of D.Coulin [3].

store all his books, music and other elements of communication, and mechanized so that it can be accessed very quickly in a flexible manner.”

Today this article looks still prophetic. A feeling still prevails that the WWW gives almost unlimited access to a kind of universal memory, which contains all the old and upto-date knowledge. The development of services and in particular the possibility to access all distributed information around the world gives the feeling that it is enough just to store links to automatically supply our knowledge, giving us a sort of magnified brain.

This view is entirely correct if we see memory merely as a storage of knowledge with the ability to retrieve them quickly. Today we understand that human memory cannot be seen as an access mechanism to information continuously accumulated without any limit. Despite its rich combinatorics, no human brain would have a sufficient storage capacity. Memory has very specific functions such as capacity of short or long term storage, abstraction, recall and, last but not least, forgetting.

Without going into the details of the involved biological processes, it is useful to assess the fundamental properties of human memory to be able to characterize exo-memory.

As summarized by G. Chapouthier [6], we can characterize human memory with three axes: sensory, temporal and abstract. The *sensory axis* includes all sensations that are tactile, auditory, visual, olfactory, etc. The *temporal axis* refers to the persistence of memory: short persistent is the working memory (at most a few minutes), also known as episodic or transient memory. More sustainable (from a few hours to several years) is the reference memory. It corresponds to the stable knowledges. Finally on the *abstract axis* are the procedural or implicit memory (acquired habits) and the explicit or declarative memory (the meanings). In the first, recall is spontaneous and immediate (for example the highly exercised sport gesture, but whose execution is made thanks to fast neural circuits), while in the second a longer reflexion is needed. This one requires the use of some form of reasoning.

So broadly categorized, each type of memory has its own mode of use or recall. For example for the implicit memory the recall is unconscious and mechanical, whereas for the explicit memory, the recall is conscious and may require application of rules. These mechanisms of recall, more or less quick, are associated with a phenomenon of forgetfulness that operates a selection in both directions on what needs to be “recorded” or not, and what needs to be recalled or not taken into account.

Here our goal is not to try to build a model of the biological memory as described above, but mainly to search for forms of mechanical extension, which could strengthen it. One way to address this issue is to retain from the biological memory the parts which can be outsourced, and possibly processed numerically.

We will therefore focus on the forms of memory corresponding to: digitized information (sensory axis), persistent (temporal axis), and conscious (abstraction axis).

There are two additional key features of an exo-memory as we see it.

- Private vs. Public: an exo-memory is a private artifact. Its isolation from the outside world and its access restricted to one subject are the key assets for the exo-memory to work. This memory is indeed useful only if the subject can enter whatever he wants with a complete feeling of liberty and security; for example storing his most secret codes.

This aspect of the exo-memory imposes some experimental limitations. This is clearly shown in [7] where it is noticed that experimentations must take into consideration legal and social aspects. Moreover, the inviolability of the support must be ensured. In the current state of technology, this is not possible⁴. One way to approach this, in particular, is to limit the exo-memory size to a small file that can be encrypted on a personal computer (or a similar artifact).

- Automatically vs. Human managed: The exo-memory cannot be limited to accumulating data obtained from all kinds of sensors attached to the subject. Initially, it can only be controlled by the subject. Only the subject can select the events he considers appropriate, introduce them and annotate them in such a way that he will be able to retrieve them even in case he has forgotten them completely. Such a task cannot be accomplished -at least on an insufficiently full exo-memory-, by an automaton, since only the subject can perform the event choices on the fly using his biological memory. Perhaps this would no longer be possible within seconds or minutes later.

3 Experimentation

We report here some experiments with an exo-memory whose management is done through a text file, used and maintained for several years using only “emacs” as text editor, the use of its function “search” as a recall function, and a single subject (the author of this paper). The description is somewhat simplified, but reflects the essential structure of the artifact and the behavior of the subject.

The basic information unit is a line (of any length), called *m-line*, *m* for “memo”. M-lines can be added one after the other or placed in the middle of others; a m-line can simply be amended. The principle is that a m-line has a single main topic. A m-line consists of strings of words or symbols separated by a comma or a separator of any kind. The m-lines are separated from each other by simply starting at the beginning of a line without any spacing. Each m-line is supposed to reflect or stimulate a memory fact useful or important for some reason.

The text is written without special formatting and all kinds of information can be inserted. If we wish to introduce information that is not textual (image,

⁴ This point deserves a further development, but it is not treated here.

sound, long document in a different coding), one introduces only the metadata (see below) and some pointer on the location of the document (personal computer space, physical domestic space or WWW pages). The search function of the text editor serves also as search engine. At any moment one can do a search to verify consistencies between different m-lines. It is important to note that the search engine is not used to retrieve an exact information, but to find some memory facts. Hence we call it “recall engine”.

The complete file is called *m-book* (for *memo-book*). Several precautions must be taken however so that the engine can operate efficiently: normalization in writing, insertion of metadata and file organization.

- **Normalization in Writing.** To be recalled without difficulty, some parts of the text should be written without misspelling⁵. Family names must be correctly written, telephone numbers must have the same shape, in short, the respect of some text norms is necessary. This standardization effort is required at least for some words which will play the role of keywords. For example it must be possible to perform a reverse lookup on a phone number. These “standards” however, may remain personal. The only reason for such standardization is to allow the subject to conduct a reverse search. He must use the standards that he knows precisely or is able to memorize on the long term (using his most stable habits).

- **Insertion of Metadata.** A metadata is a piece of text which will help to recall a m-line. A wide discretion is left to the form of metadata that can even be formulated in an incomprehensible or unstructured language. This data can be put anywhere in a m-line and is generally redundant. For example, to recall the name of a person one can seek it with the first name first or with the family name. One cannot usually predict what will come first to mind, especially if one has been without any relationship with this person for many years. If a m-line deals with **Pierre Deransart** one can set a metadata as **Pierre Deransart pierre** or **deransart pierre Deransart**.

More generally, one can put in the metadata context information that can help to find the memory fact by using other information. For example, one can enter:

```
Pierre Deransart beardless pierre clear eyes
sport pierre sport Logic Programming ICLP
Deransart lp ....
```

The principle is to put several peculiarities, trying to imagine with which words **pierre** could be retrieved after several years, after having forgotten all about him.

- **File Organization: Building m-Paragraphs and m-Pages.** As the size of a m-line increases it becomes more difficult in practice to read the information it contains. It can then be better to split it into several m-lines. For example, after a few years, one has accumulated several types of information

⁵ At least with a spelling that is consistent for the subject. With a modern recall engine including some grammatical treatments, such a requirement could be avoided.

about a person such as location(s), contact means (email, phone, fax, ...), family composition evolution, meetings of interest (cultural events, debates ...), jobs, adventures, pictures, publications,..., which it was felt that could be useful to keep the memory without creating any particular medium of archiving. A better readability can be thus obtained by splitting the existing m-line into several m-lines with a better thematic homogeneity. These m-lines are separated by “new-line” characters. This new set of m-line without spacing is called a *m-paragraph* and is devoted to a particular topic. A m-paragraph may change over time, resulting in enrichment of size and numbers of m-lines. From time to time, if the topic is growing, new paragraphs or new m-lines can be created by extension or by splitting.

Here is a non compromising and relatively understandable example of a short paragraph with m-lines and metadata about the management of a coffee machine in the research team on “constraints” at INRIA where the subject is working.

```

CAFE inria cafe machine cafe inria cafe depannage cafe
contraintes cafe projet cafe commande cafe
-Societe D8, 7-8 rue Leon Geffroy, 94408 Vitry Cedex
-Client 30700
-commande: 01 47 18 38 40, par 200 pour 70 euros, 3 cat:
fort (brun), moyen (vert) et faible
-depannage: 01 47 18 38 30 7h30-17h (9h samedi)
-matricule appareil: 034726
-commande 9/4/10: 35 euros (cafe 100 doses, gobelets, spatules
et sucres, sinon 32 euros) livraison a l'occasion (a partir du
lundi 12)

```

The first m-line sets the main topic. The spelling is simplified following subject rules (lack of accents); the m-lines are very short and relate some data concerning the machine, incidents or orders. The metadata of the first m-line was introduced to improve previous painful researches of the m-paragraph and correspond to different possible combinations of main topic and contexts: “coffee inria” (workplace) or “coffee constraints” (project team) or “coffee project” or “coffee order”.

Since there is no growth strategy of the exo-memory, several m-paragraphs on similar topics may have been created. It is then possible to group them in order to facilitate the understanding of the general topic. Here the physical medium is reaching its limits as it is used, because of course many combinations are possible and this support is not intended to allow all sorts of groupings. This point will be discussed later. This grouping is purely casual, but it can also be obtained by adding metadata to identify a new set of m-paragraphs and to allow to scroll through the m-paragraphs corresponding to this new topic. Such a grouping is called *m-page* and there are likely several kinds of possible m-pages. The final decision is left to the subject whether to create a new m-page, or rearrange m-pages.

- **Treatment of obsolete data.** Some data may become obsolete (e.g. change of address of a friend), some are just uncertain (name misheard over the phone or age of a person not known with certainty). Since the data are not formally transformed, but just selected by the subject when trying to recall them, the single symbol “?” is used for uncertainty, and “%” for obsolete data. The prefix operator “%” applies to the entire sentence. The prefix or suffix operator “?” applies to a word.

The obsolete data remain stored and sometimes moved at the end of a m-paragraph for better readability. It may be interesting to keep them. A question still concerns the complete erasure of data (for example in case of change of an URL). Such a deletion is rare, because keeping an obsolete data does not disturbing in general neither recall nor readability.

Evolution of the m-book therefore develops the following (non exhaustive) action types. For the m-lines: creation (*create_m-line*), insertion of data or meta-data (*insert_m-line_content*), linguistic corrections, marking dead data (*dead_content*) or uncertain (*uncertain_content*); for m-paragraphs: creation, split or combination of m-lines (*create_m-paragraph*), fusion (*fusion_m-paragraph*) or split (*split_m-paragraph*) of m-paragraphs; for the m-pages: creation by grouping of m-paragraphs (*create_m-page*) or by adding metadata in the m-paragraphs which compose it, grouping of m-pages (*fusion_m-page*).

Note that we could increase the number of higher level structures indefinitely (m-chapters, ... etc.). The exo-memory, as implemented here, makes it difficult to go beyond two levels and this is unsatisfactory since the creation of a m-page may disrupt another one. In practice the multiplication of levels is not really useful, because it would amount to impose an overall structure on the m-book. Moreover there is no certainty that this structure retains the same consistency over time since the subject evolves and his memory as well. Allowing different possible m-paginations would be useful.

We have tested for 6 years this approach with a private text file which reveals to be an exo-memory in practice very useful, fast, efficient and overall persistent in the sense that the recall engine does not speed down over the time. The strategy that avoids this degradation and, on the contrary, that continually improves the effectiveness of the recall (at least for frequently consulted facts), consists in consistently extending metadata, each time a m-line or a m-paragraph is not immediately recalled, with a few keywords or a single expression. Such new metadata cannot be automatically inferred since the words or phrases that one would like to add (as a form of mnemonic shorthand) are terms that come out from the subject’s memory and are often unpredictable.

Through continuous use, the growth of m-book has been around 200 KB per year. This growth is linear and not exponential, since the introduction of new information is made exclusively manually, in such a way that the size of the inputs remains proportional to the average time needed to introduce them. The size is also limited by the fact that, even if it is possible to introduce portions of

text using copy and paste, generally only pointers or references are introduced in case of voluminous data. It is essential for the m-lines to be as short as possible in a m-paragraph. This shows that during a whole life the size of the m-book (of the order of several tens of MB) cannot be a real obstacle to the efficiency of many possible applications and services, particularly of the recall engine.

The growth speed would be different if the m-book was built automatically, for example from a personal ontology reflecting potential personal interests. This would only reflect the exponential growth of global knowledge, but corresponds to a different problem. It would still be necessary to make a data selection in order to retain only the information sufficiently reliable and significant from the subject point of view. This can only be achieved by the subject.

Along the years, m-paragraphs and m-pages are created, completed and re-organized. The ones which are more frequently used are easily recalled. Interestingly but not surprisingly there is constant need of completion of metadata. However as long as these most frequently used m-pages are consulted, recalling them is easier. For m-paragraphs or m-pages which are not consulted at all after built, the recall may be costly, but rarely fails. It may be observed that m-pages evolve as current interests of the subject are evolving too.

4 Modeling with Traces

We briefly present a possible theoretical approach based on the notion of trace as presented in [8–10] and inspired by software engineering.

The main idea is that the m-book as described above is one of the several possible representations of the state of a system (the “memory”) which results from a serie of events called *trace*. This trace can be formalized by a so called *actual trace*⁶. The state of the memory at a given time, whose m-book is a visual possible presentation, is said *virtual state* of the memory. At time zero, the *initial* virtual state, as the memory, are assumed to be empty. The *current* virtual state (beyond the initial state) can be fully known using the actual trace only.

The semantics associated with such a trace is a semantic of reconstruction or *interpretive semantics* (IS) that allows to reconstruct a current virtual state from the actual trace. The IS is given here with the trace. It allows to interpret the actual trace⁷ by representing it in the form of a m-book. It is important to distinguish the interpretation of the trace as an abstract data structure (parts of text ordered by a system of pointers) and the various representations it may have. Here we limit ourselves to a two-dimensional representation in the form of m-book, but one could imagine applications that perform more sophisticated representations including several possible dimensions and offering the possibility of several m-paginations.

⁶ Contiguous integral actual trace in [8].

⁷ For this reason, it is called *interpretive semantics* in [8].

In this approach one can consider that any m-line, any m-paragraph or m-page, . . . , is a sub-trace. The m-book can thus be seen as a trace base system in the sense of J.-C. Marty and A. Mille [11, 12] whose the actual trace described here forms a *primary trace* (“trace première”). This opens for using learning methods in a theoretical well founded framework to design tools helping the subject to use his exo-memory to assist him in discovering new knowledge based on his own experience.

5 Discussion

We discuss this model of exo-memory and compare it with works on biological memory, knowledge engineering and existing note organizer softwares.

In a neuro-biological approach G. Edelman and G. Tononi [13] pointed out that memory is essentially non-representational⁸ and that the brain is actually filled with hundreds or even thousands of memory centers in constant interactions. Our exo-memory enables us to maintain in a persistent way pieces of memory, allowing to construct and to maintain all kinds of personal semantic networks. But it has some limitations. On what we have called the sensory axis the introduction of a new information may be related to several sensations. These may correspond to the context of an event which may contribute to its recall, but will not be stored in the exo-memory because this would take too much time or because this kind on influence is unconscious. For example, in the experiments of [2], the influence of the color of a document or the weather at the time of recording a fact are stored. Such information is rarely noted with our type of textual exo-memory, mainly because, at the moment of writing it, such factors are unconscious.

Somehow the exo-memory focuses on the abstraction axis, hence handling conscious acts and relations only. It is likely that the act of introducing some information in the exo-memory helps by itself to its memorization and contributes also probably in the unconscious part of it. But it also has a constraining aspect which may contribute to restrict the proper act of storing memories.

In the field of medical assistance to patients with recognized memory deficits [14], the introduction of memory facts, .i.e the selection of relevant facts, is done with the help of staff carer or a relative. The problem, as the results one can expect, are essentially different.

Researches in knowledge engineering are better oriented towards social memory, i.e. the storage and sharing of knowledge within groups of people as diverse as family, business, social network or even the entire humanity in the globalization context. So in the Handbook of Research on Emerging Rule-Based Languages and Technologies [15] most of the related works concerns the automatic construction of ontologies for the management of archives in various social contexts.

⁸ This means in particular that we do not memorize all details of a scene, but only a few elements used to partially reconstruct it according to specific needs.

These approaches frequently include semantic network or memory, as introduced by Quillian in 1968 [16]. It is important to observe that some network can be built from the virtual memory represented in the physical medium. It could serve as a basis for several applications that facilitate access to and management of the exo-memory. But in order to preserve the plasticity of the exo-memory, such network should be built and used only gradually, and especially cannot be imposed a priori.

The case of software designed to manage notes, if not all carriers of the subject's activities, as EverNote, DevonThink, CintaNotes, SOHO Notes 8 Yojimbo, ShoveBox or wikidPad, among others, correspond to an intermediate situation where the aim is to facilitate the organization not only of a few notes related to the subject's life, but to some extent of all documents to be manipulated. These programs usually require a structure that, at some stage of development, may become constraining. We can not here discuss each system, but our approach has a serious advantage: the simplicity of the needed software to run it. Indeed many proposed systems are extremely sophisticated and the user may become dependent on some supplier. By using a system whose functions can be reduced to a "single" text processing, the subject can be pretty sure that he/she may use his exo-memory on the long term.

In our approach, personal information storing on digital media actually has two parts: exo-memory and personal archives (the collection of all documents stored by the subject). The exo-memory, reduced to a single file, can act as a gateway to help to find archived documents, playing the role of the thesaurus of an encyclopedia. On one hand the exo-memory must have an exclusively private status (private property and exclusive access by the subject) and its way of handling reduces its growth; on the other hand, the status of the archives is necessarily different because of their mode of growth that can be shared, automated and exponential. Thus its private status and relatively self-sufficient semantics may not be guaranteed at all. Thus we see that there are two distinct areas of research and the respective related works, while retaining some common topics, are of different nature.

6 Conclusion

We have shown how a consciously written trace including as many spontaneously selected or thought about life facts as possible, that are tirelessly recorded and organized by its subject, could be an exo-memory. We have shown how a very coarsely structured text file that is manipulated using a text editor, could constitute a useful approximation, taking into account some plasticity aspects of the neuronal memory. Finally, relying on the observation that the sequence of updates is a primary trace, we found that this approach allows the development of utilities liable to improve the performance of the exo-memory thanks to interfaces that make it easier to use. Many improvements are indeed becoming possible, based on combining existing tools, in particular, in the fields of

databases, data mining, abstract interpretation and information retrieval. Such applications are partly included in the note organizer softwares we have quoted.

The originality of this approach lies in the essential characteristics of this form of exo-memory: simplicity and efficiency, mnemonic and creative functionalities, and feeling of satisfaction. This last point is particularly important. To the extent where the subject very often feels that he recovers “memory” thanks to this tool, or at least he does not lose it, its usage induces a positive reinforcement to use it even more. Provided that this feeling is not counterbalanced by the difficulty to use the tool, the lost time to enter data is clearly outweighed by the time saved in recalling them.

The first point (simplicity and efficiency) is also essential for use by a non-specialist of keyboard, mouse and a text editor, but it is also a guarantee of independence and long term availability of the exo-memory. The accessibility is probably a major technical difficulty to overcome. However other input methods such as audio input or handwriting could be adapted to facilitate exo-memory management. With regard to the mnemonic and creative features, they are guaranteed by the voluntary act of choice of the relevant events which accompanies and enhances the activity of the biological memory of the subject. The memorized events are not selected by an automated process, but chosen and adapted by the subject who is himself changing all the time.

Finally, we insisted that exo-memory must remain completely private and that only information chosen by the owner may be communicated outside (it is indeed the case of the biological memory). An exo-memory has no other social function than to provide assistance to its owner.

The approach presented here is more like a working tool on oneself [17], notepad or personal hypomnema. If it is true, as asserted by Michel Serres [18], that the new technological means generate forms of neo-Darwinism⁹, such new facilities should also help in fostering a work on oneself, always intimate and essential, while benefiting from technological advances.

Aknowledgments

I thank the referees for their interesting and inspiring remarks.

References

1. Delahaye, J.P.: Complexités. Aux limites des mathématiques et de l’informatique. Belin - pour la science (2006)
2. Fuller, M., Kelly, L., Jones, G.J.F.: Applying contextual memory cues for retrieval from personal information archives. In: PIM 2008 - Proceedings of Personal Information Management, Workshop at CHI. (2008)

⁹ Basically, the idea is that modern storage capacities of Information obviate the need to concentrate efforts on pure human cerebral memorization, and can thus allow to release new functionalities of the brain.

3. Coulin, D.: Les traces. Editions Bernard Grasset, Paris (2004)
4. Deransart, P.: Towards a Simple Textual Trace Based Personal Exo-Memory. Technical report, Inria Paris-Rocquencourt (septembre 2010) <http://hal.inria.fr/>.
5. Bush, V.: As we May Think. The Atlantic Monthly (July 1945) The electronic version was prepared by Denys Duchier, April 1994, <http://ccat.sas.upenn.edu/~jod/texts/vannevar.bush.html>.
6. Chapouthier, G.: Biologie de la mémoire. Odile Jacob (February 2006)
7. Vemuri, S., Bender, W.: Next-generation personal memory aids. BT Technology Journal **22**(4) (October 2004)
8. Deransart, P.: Conception de Trace et Applications (vers une méta-théorie des traces). Technical report, Inria Paris-Rocquencourt (march 2009) Working document <http://hal.inria.fr/>.
9. Langevine, L., Deransart, P., Ducassé, M.: A generic trace schema for the portability of cp(fd) debugging tools. In Apt, K., Fages, F., Rossi, F., Szeredi, P., Vancza, J., eds.: Recent Advances in Constraints. Number 3010 in LNAI. Springer Verlag (May 2004)
10. Deransart, P., Ducassé, M., Ferrand, G.: Observational semantics of the resolution box model. In Vanhoof, W., Hill, P., eds.: Proceedings of the 17th Workshop on Logic-based Methods in Programming Environments (WLPE'07), a post-conference workshop of ICLP'07, Porto, Portugal (September 2007) à paraître dans le **Computing Research Repository (CoRR)**.
11. Marty, J.C., Mille, A.: Analyse de traces et personnalisation des environnements informatiques pour l'apprentissage humain. Hermès, Lavoisier (2009)
12. Mille, A.: From case-based reasoning to trace-based reasoning. Annual Reviews in Control **2**(30) (2006) 223–232
13. Edelman, G.M., Tononi, G.: A universe of Consciousness. How Matter becomes Imagination. Basic Books (2000) French translation: "Comment la matière devient conscience", Odile Jacob, 2000.
14. Matthiew, L.L., Dey, A.K.: Lifelogging memory appliance for people with episodic memory impairment. In: Proceedings of UbiComp'08, Seoul, Korea, ACM (September 2008)
15. Giurca, A., Gasevic, D., Taveter, K.: Handbook of Research on Emerging Rule-Based Languages and Technologies, Open Solutions and Approaches. Information Science Reference (2009)
16. Collins, A.M., Quillian, M.R.: Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior **8**(2) (1969) 240–248
17. Foucault, M.: L'écriture de soi. Dits et écrits **4**(329) (1984) 415–431 1980-1988, 912 pages, 140 x 225 mm. Collection Bibliothèque des Sciences humaines, ISBN 2070739899.
18. Serres, M.: Les nouvelles technologies, que nous apportent-elles? Interstice (2006) Michel Serres's conference recorded at "Ecole Polytechnique", decembre 1rst, 2005.

LinksTo – A Web2.0 System that Utilises Linked Data Principles to Link Related Resources Together

Owen Sacco¹ and Matthew Montebello¹,

¹ University of Malta, Msida MSD 2080, Malta.
{osac001, matthew.montebello}@um.edu.mt

Abstract. Although social sharing websites are currently employing Semantic Web techniques to structure the data, these websites still stand in isolation since most of the data is not linked. Therefore, this paper proposes a prototype system called *LinksTo* that provides users the functionality to link resources from the Web at large. These links are described in RDF adhering to vocabularies recommended by the linked data best practices. The links described in RDF are transparent to the user. However, the system also provides functionalities to access the RDF data that can be utilised by the linked data community.

Keywords: Semantic Web, Linked Data, Web2.0, RDF, Ontologies, FOAF, SIOC, SKOS.

1 Introduction

With the advent of Web2.0, many social collaborative platforms have emerged providing users the functionality to share information in a personal and collective manner. Most of these social platforms consist of wikis, blogs, social bookmarking websites, photo sharing websites and video sharing websites. The majority of these community systems also provide the functionality to organise and describe the content by means of a lightweight knowledge representation called folksonomy that consists of describing the act of tagging Web resources. Although some of these Web applications are utilising Semantic Web technologies to add more meaning and structure to the data, most of the data is not structured using linked data principles and practices [1]. Moreover, even though Semantic Web technologies are being exploited in current collaborative knowledge sharing web sites, the data of such web systems is still not linked and therefore the data is still isolated limiting the aspect of collaborative knowledge sharing amongst web systems. Therefore, this necessitates for a system to allow users to link resources from diverse web systems and such links are described according to the linked data best practices.

The aim of this paper is to propose a prototype of a Web2.0 system called *LinksTo* that utilises Semantic Web technologies whereby various resources can be linked and such links are described in RDF conforming to vocabularies recommended by the linked data best practices. LinksTo provides an interface for users to collect and link resources that are related to a particular topic the user is searching on that would collectively form a collaborative sharing of information.

This research paper is organised as follows: in section 2, a brief discussion about related work is provided. Section 3 provides a brief explanation of the functionality and the technical aspects of the LinksTo system. In section 4 a concluding note is provided that summarises this research paper.

2 Related Work

Social resource sharing systems consist of web-based platforms that provide the users to publish, share and manage resources. Such systems have attracted a number of users because these platforms do not require any specific expertise. One of the most popular social resource sharing systems are the social bookmarking websites such as delicious¹ and BibSonomy[5] that assist users to save, share and tag URLs of resources for later retrieval. Resources which are bookmarked in these social websites are organised and indirectly linked by means of tags. Linking of resources is achieved by assigning the same or similar tags to each resource's URL. Therefore this implies that if users want to link resources, the users have to save related resources one at a time rather than as a bundle of resources and also have to assign the same tag(s) for each individual resource. This could be a daunting task if the user wants to link a large number of resources. Moreover, current social bookmarking websites do not provide the functionality to describe linked resources and their assigned tags in RDF adhering to the linked data best practices. Hence, LinksTo is designed to extend the idea of social bookmarking websites by providing users the functionality to save multiple resources at one instance, assign tags to the collection of these resources and describe in RDF the linked resources together with their assigned tags.

Apart from social bookmarking websites, Baeza-Yates and Tiberi [3] propose how to extract semantic relations from query logs. The authors explain how query logs can be presented as a folksonomy whereby user queries act as tags assigned to documents clicked by the same user after the query result was retrieved. This method shows positive results for information retrieval, however, this method does not take into consideration that clicked documents may not be relevant to what the user requires. It is the norm that the relevance of a document to the topic the user is searching on is known after the user has examined the document. Therefore, relying on clicked documents is not sufficient in order to link related resources. Another similar approach is proposed in [6] whereby the authors term logsonomy as a folksonomy for web search engines. Their approach is to define a folksonomy for the relation between a query, a clicked document and the user. However, since this approach is also based on clicked documents and disregards the relevance of the content of a document is to a query, this approach does not yield accurate results when linking resources.

An approach that is similar to the LinksTo application is the GroupMe! system [7]. GroupMe! provides a Web2.0 interface for users to group resources and also to tag such resources. The GroupMe! system also describes in RDF the grouped resources and their associated tags. Moreover, this system extends the tripartite folksonomy tuple defined in [5] by adding a concept called a group that denotes the grouped

¹ <http://delicious.com/>

resources. In order for the GroupMe! system to describe the grouped resources in RDF based on this extended folksonomy, a GroupMe! specific ontology is used. The problem with this specific ontology arises when external applications want to make use of the RDF data. This is because when applications use diverse system specific ontologies, application developers implementing systems that consume RDF data have to be aware of all these ontologies in order to exploit the RDF data. If the application is not aware of the specific ontology in use, then the application will not be able to parse such RDF data. In fact, in order to publish data on the Web, the authors in [2] recommend that vocabularies such as Semantically Interlinked Online Communities (SIOC), Friend-Of-A-Friend (FOAF) and Simple Knowledge Organisation Systems (SKOS) are used wherever possible in order to simplify client applications to process the data. Therefore, LinksTo models its data on the tripartite folksonomy model defined in [5] rather than extending it and uses the SIOC ontology to describe in RDF the linked resources. This allows the data to conform to the linked data practices allowing other applications to utilise the data without the need to conform to any system specific ontology. Furthermore, LinksTo provides other features that are not present in GroupMe! such as: users can follow other user's linked resources; users can edit tags when editing linked resources; users can link their user profile to other social network profiles of the same user; and the user profile is described in RDF using the FOAF vocabulary which can be exported to an external file or linked from other Web systems.

3 The LinksTo System

3.1 A Web Search Scenario – LinksTo Motivation

A user is searching on a particular topic and the web search engine retrieves many resources that are related to that specific topic. However, only some of the resources are important and relevant for the user. Once the user has decided which resources are relevant, the user would then desire to save all the significant URLs for later retrieval. Since the resources are all related to the topic the user was searching on, the user would require linking the resources and tagging this collection of linked resources with keywords that describe best that specific topic. Therefore, the user adds all the relevant URLs to a space that links these resources and such space also provides the user with the functionality to assign tags to the linked resources. The user or any other user(s) that eventually would want to search on that same topic can query the system by using tags; and the linked resources tagged with the exact or similar tags can be retrieved. With this approach, the user searching for resources will be presented with the most relevant resources and the user does not need to filter out unrelated resources. Thus, the user only searches for a desired topic and the linked resources are displayed as the search results.

3.2 LinksTo User Features and Functionalities

In a nutshell, the LinksTo system provides the following user functionalities:

1. Creation of user profile. LinksTo provides users to create a profile that can be linked to other profiles of the same user created in other social network systems. Moreover, if the user has a FOAF file, this can be uploaded by the LinksTo system. Once the FOAF file is uploaded, the system parses such FOAF file and retrieves any data that is required within the system.

2. Creation of Web resources *Links*. LinksTo provides users the functionality to create links amongst various Web resources and collect them in a *Link*. This functionality also extracts any available tags assigned to the resources found in delicious. Once the tags are extracted, the system adds them to a *Link* tag cloud which consists of a set of tags assigned to a particular *Link*. The system also allows users to add or remove tags from the *Link* tag cloud. Once a *Link* is saved, the Web resources links are described in RDF using the FOAF, SIOC and SKOS vocabularies and the data stored in persistent storage. Fig. 1 depicts a screenshot of the creation of *Links*.

3. Searching mechanisms for Web resources and *Links*. LinksTo provides searching capabilities for Web resources using the Google search API. When a Web resource is selected after retrieved by the search engine, *LinksTo* offers the option to preview the Web resource within the system. Moreover, LinksTo provides searching functionality using tags as queries to search for *Links* within the system.

4. Exporting RDF data. The system provides users to export RDF data to an RDF file. Moreover, the system also provides application clients to request for RDF data in the form of HTTP requests and the system sends back the linked resources structured in RDF. This is convenient for applications that utilise Semantic Web technologies since such applications can make use of the LinksTo's data without the need to re-format the data.

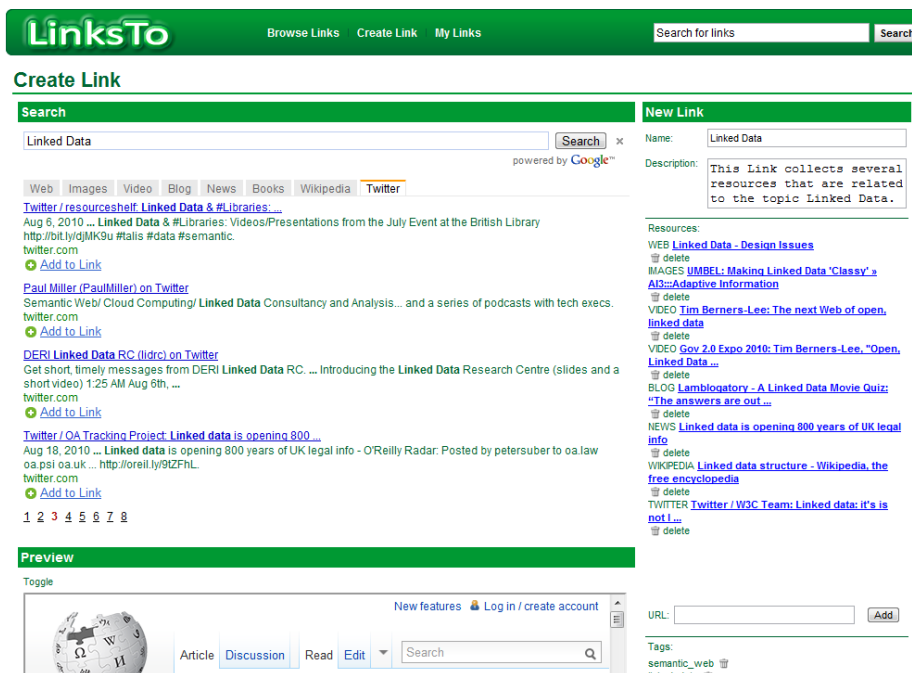


Fig. 1. A screenshot of LinksTo creation of a *Link*

3.3 LinksTo Architecture – A Technical Overview

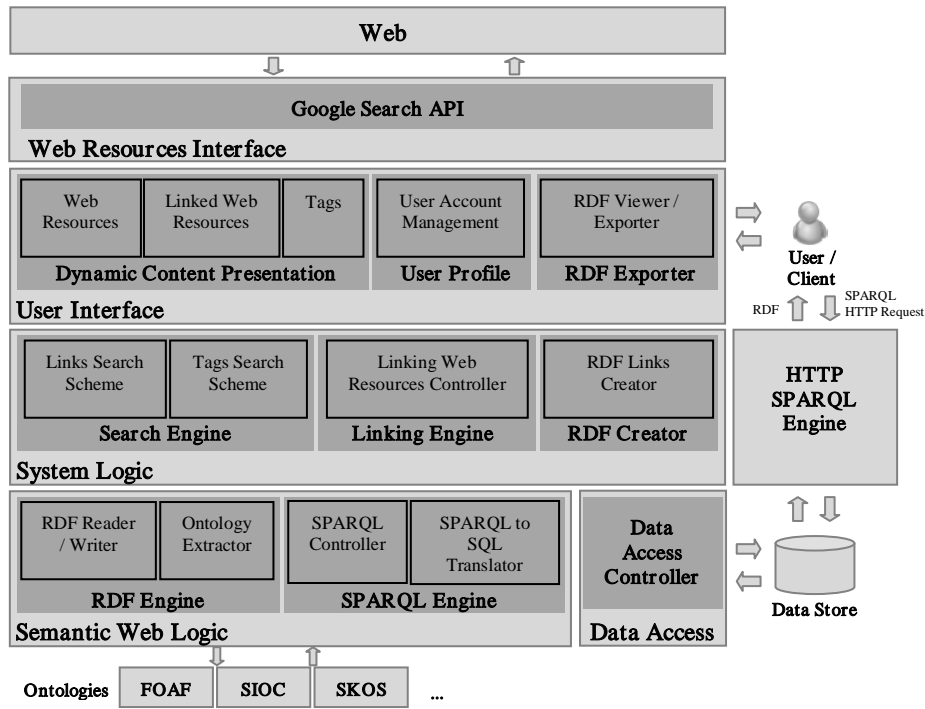


Fig. 2. LinksTo Technical Architecture

LinksTo technical architecture consists of various specific functions as illustrated in Fig. 2. These functions are briefly explained below.

Web Resources Interface. This module interacts with the Web resources located on the Web. This module contains the Google search API that retrieves search results of Web resources using the Google search engine. Moreover, this module also retrieves tags (if available) assigned to the resources from within social bookmarking websites such as delicious.

User Interface. This module provides a user interface for: 1) searching Web resources on the Web; 2) previewing selected Web resources from within the same application; 3) to create *Links* and tag such *Links*; 4) to submit search queries to search for *Links*; and 5) an interface to manage user account information. The User Interface uses the Prototype framework to perform AJAX calls to retrieve data.

System Logic. The system logic module controls most of LinksTo technical functionality. This module contains the *Links* searching engine that consists of the mechanism to query and retrieve the relevant *Links* which are the closest to the user's query terms. The searching engine makes use of a ranking procedure called FolkRank [4] to rank the tags assigned to the *Links* and the ones with higher ranks are displayed as top results. This ranking procedure was used since the results stated in [4] have proved to be adequate. Although the FolkRank ranking algorithm was used as a

preliminary ranking strategy in LinksTo, in the near future other ranking algorithms will be studied and the one which provides the optimum results will be used. Apart from the searching engine, this module also contains a controller that controls the creation and amendments of *Links*.

Semantic Web Logic. This module is responsible for parsing and writing RDF statements, and for querying RDF models. The Jena framework² is embedded in this module since it provides the functionalities to: 1) read and write RDF statements; 2) a SPARQL engine that queries RDF models and 3) methods for in-memory and persistent storage. Moreover a REST web service is created to interact with Jena that acts as an HTTP SPARQL engine that allows SPARQL queries to be requested over HTTP and sends the results back to the client application.

4. Conclusion

The LinksTo prototype system provides a Web2.0 interface utilising Semantic Web technologies that provides users with the functionality to link Web resources. This system also provides Semantic Web developers with data formatted in RDF that can be reused. Since LinksTo is still in its infancy, once the *Links* dataset increases, this dataset can be used to analyse user behaviour with respect to how users link Web resources. This analysis can contribute to interesting research as to how to retrieve information based on linked resources. Moreover, LinksTo system adds value to the Web community by promoting the use of Semantic Web technologies adhering to the linked data practices in order for the Web of Data to continue to evolve.

References

1. Berners-Lee, T. (2006). Linked Data - Design Issues. [Online] <http://www.w3.org/DesignIssues/LinkedData.html>
2. Bizer C., Cyganiak R., Heath T. 2007. How to publish Linked Data on the Web. [Online] <http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/>
3. Baeza-Yates R., Tiberi. 2007. Extracting Semantic Relations from Query Logs. In KDD'07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, p. 76-85.
4. Hotho A., Jäschke R., Schmitz C., Stumme G. 2006. Information Retrieval in Folksonomies Search and Ranking. In: The Semantic Web Research and Applications, p. 411-426.
5. Hotho A., Jäschke R., Schmitz C., Stumme G. 2006. BibSonomy: A Social Bookmark and Publication Sharing System. In: Proceedings of the Conceptual Structures Tool Interoperability Workshop at the 14th International Conference on Conceptual Structures, p. 87-102.
6. Krause B, Jäschke R., Hotho A., Stumme G. 2008. Logsonomy – Social Information Retrieval with Logdata. In HT'08: Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia, p. 157-166.
7. Abel F., Frank M., Henze N., Krause D., Plappert D., Siehndel P. 2008. GroupMe! – Where Semantic Web Meets Web2.0. In: The Semantic Web p. 871-878.

² <http://jena.sourceforge.net/>