Smarter Groups – Reasoning on Qualitative Information from Your Desktop

Sebastian Böhm, Marko Luther, and Matthias Wagner

DoCoMo Euro-Labs, Landsbergerstr. 312, 80687 Munich, Germany
`<lastname>@docomolab-euro.com`

Abstract. This paper explores the possibilities and core technologies of the ongoing development towards the Semantic Web for desktop application enhancement. It demonstrates how an ontology-based software can provide refined support for personal information organization. The Web Ontology Language and reasoning mechanisms based on Description Logics (DL) are analyzed as enabler technologies for semantic enrichments. We report on our prototype that realizes the enhancement of Apple's Address Book application by DL-based smart groups.

1 Introduction

The communication environment we find today dramatically changed the way we work and interact with other people. Access to common knowledge sources like the World Wide Web and means of communication like E-Mail have long become ubiquitous and intensified the need for more efficient and intelligent knowledge management mechanisms. In this respect, creating new approaches for an integrated information management and distribution is one key task the IT industry and the computer science community are currently dealing with. Therefore, a lot of effort is put into the development towards the Semantic Web [1] to provide a machine readable and meaningful description of the elements of the World Wide Web. Certainly, desktop applications could also profit exceedingly of well-founded logical annotations, which enable adequate techniques for handling distributed data more efficiently. Based on semantic descriptions, such applications would comprise additional reasoning mechanisms and therefore outmatch ordinary database driven approaches. In this context, we discuss an experimental approach for information management by establishing a link to technologies of the Semantic Web. To achieve this, a custom built set of ontologies as well as a prototype for ontology-based desktop application enhancement will be introduced.

2 Ontologies

Realizing interoperability between different knowledge representations, a set of core ontologies written in the Web Ontology Language (OWL) has been developed to model a concrete use case for application enhancement while at the same time providing a linkup to common upper context ontologies. The core ontologies altogether consist of nine components, defining more than 300 concepts and nearly 250 properties. Each of these core ontologies contains a specific vocabulary concerning the representation of the domain of interest that can be used for further domain specific deployment.

The agent ontology is informed by the FOAF vocabulary [2] and allows the specification of relationships between people rather than just describing a person's contact information. It covers domain independent facts about agents and provides a common vocabulary to express relationships between people in more detail. The hierarchical structure of the T-Box constitutes the framework for describing persons, groups and organizations. Social relationships between instances can be expressed by using a variety of properties, which are themselves structured in a hierarchical order with respect to source and domain restrictions.

Complementary to the agent.owl ontology the calendar.owl ontology has been modelled to consider additional facets for ontology-based information management through the description of events. Those descriptions basically consist of the date and time when the event takes place and the people attending it. The fundamental design of the T-Box differentiates between a calendar for private events, and a second one for business purposes. Further distinctions in the structure of the ontology are made according to spatial, temporal and social circumstances of events.

3 Qualitative Reasoning

Having developed appropriate core ontologies, knowledge retrieval and information management are key areas of application that profit from knowledge sharing through seamless interoperability as well as profound reasoning support. With reference to databases of personal contacts, for instance, several other facts and assumptions can be derived by making use of ontology-based knowledge management and inference.

Thanks to the logical coherence of ontologies, specific information can be derived based on the transitivity and reflexivity of certain entities and general classification. This additional information is consequently used to complete missing data in contact databases. Figure 1 illustrates some of the social relationships of a person called Dawson Campbell. It should be noted that even if the relationship between Helen Buchanan and Dawson Campbell has not been explicitly defined, it is possible to draw the conclusion that Helen is Dawson's mother in law: Helen, in contrast to Dawson, is female (because she has been defined as Marks's wife) and has the daughter Madeleine which is herself the spouse of Michael. So, an additional and formerly unknown relationship between two entities can be established within the ontology based on the explicit represented information. The necessary reasoning steps make use of the world knowledge encoded in the ontology. The concepts relevant to the derivation described above are defined as follows.

$$
\begin{aligned}
\texttt{Mother\_in\_law} &\equiv \texttt{Parents\_in\_law} \sqcap \texttt{Women} \\
\texttt{Parents\_in\_law} &\equiv \texttt{Person} \sqcap \exists\, \texttt{child.Spouse} \\
\texttt{Woman} &\equiv \texttt{Person} \sqcap \texttt{female} \in \texttt{gender} \\
\texttt{Spouse} &\equiv \texttt{Person} \sqcap \exists\, \texttt{spouse.Myself}
\end{aligned}
$$

DL- based reasoning of the kind just sketched is rather limited. For example, it does not allow to draw the conclusion that Laurie and Michael are siblings. A concept defining siblings would need to use variables which are not part of the OWL DL. However, by using a rule language on top of OWL, such as SWRL [3], properties such as sibling can be defined easily.

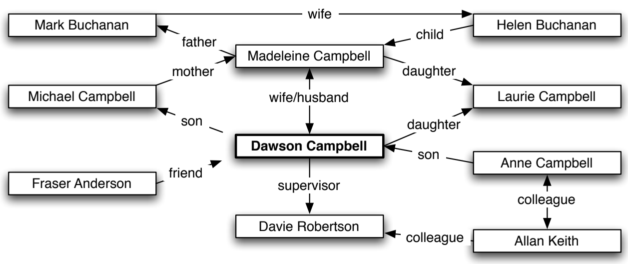

Fig. 1. Social Relations

$$(x\ \texttt{child}\ z) \wedge (y\ \texttt{child}\ z) \wedge (x \neq y) \Rightarrow (x\ \texttt{sibling}\ y)$$

By feeding a rule inference engine with rules that describe properties by property chaining, such as *grandchild*, *uncle* or the *sibling* property stated above, additional facts can be derived. In doing so, implicit knowledge is again used to derive explicit information which might trigger further derivations using the standard DL-based mechanisms.
Despite rather social relationships, concepts can moreover be described using a variety of different qualitative attributes.

$$\begin{aligned}&\texttt{Adult} \equiv \texttt{Grandparent} \sqcup (\texttt{Person} \sqcap (\exists\,\texttt{child}\,(\texttt{Student} \sqcup \texttt{Employee})) \sqcup \\ &\quad (\exists\,\texttt{degree Degree}) \sqcup (\exists\,\texttt{employer Person}) \sqcup (\exists\,\texttt{head Organization}))\end{aligned}$$

The information gained through the design of concepts like *Adult* allows for manifold usage in different areas of applications. To give an example, an ordinary desktop application like an email client could take advantage of the information made available in adjusting the composition of email messages with respect to the recipient. Thus, the application is able to differentiate between your 16-year old son and your business partners to adjust a variety of predefined settings.

4 Introducing McAnt

McAnt is our first prototype, which has been developed to demonstrate the use of ontology reasoning support for desktop applications. McAnt retrieves qualitative information from standard personal management applications bundled with Apple's Mac OS X, the Address Book and iCal. This information entails knowledge that can be derived through ontology-based reasoning mechanisms and in that respect serves as the basis for further application enhancements.

4.1 Components

The McAnt system consists of a number of linked components (cf. Figure 2). The McAnt application itself, which has been developed as a Java application with a native Mac OS Cocoa interface, the OWL reasoning engine Racer [4], Apple's Address Book and iCal as well as the set of core ontologies described above, providing relevant concept descriptions including the definitions of agents, spatial and temporal entities.

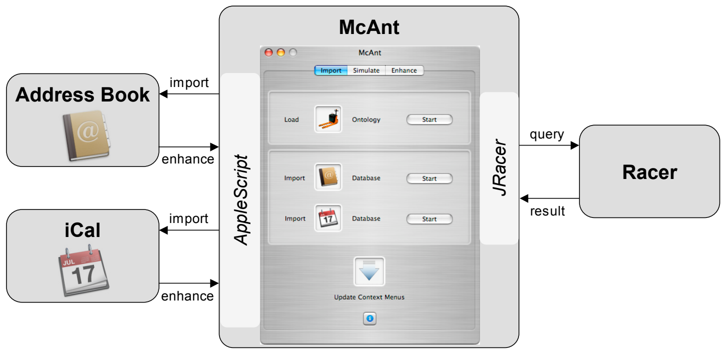

Fig. 2. McAnt Architecture

Serving as a bridge between the different components by relying on inter-application communication, McAnt accesses the Address Book database and translates it into appropriate Racer commands. The reasoning engine is accessed via JRacer, a TCP socket-based Java client for the RACER system.

4.2 Retrieving qualitative information

The retrieval of the Address Book entries has been implemented using AppleScript, a scripting language developed by Apple, which provides automation support for various complex tasks, including inter-process communication. The information stored in the Address Book comprises not only a contact database, but additionally offers the possibility to define labels describing relationships among entries including the owner.

Based on the relationship definitions, individuals that are linked with object properties are created. This enables instance reasoning support through RACER, resulting in an automatic classification within the T-Box. Thereupon the contacts have been classified and assigned to the appropriate concepts like *Family* or *Colleague*.

Furthermore, McAnt also makes use of the personal information managed with iCal. Similar to the Address Book imports, an AppleScript accesses the iCal database and translates it into appropriate RACER commands. This time the high level information retrieved consists of events, which can be associated with people attending this event and the location where it takes place. According to the concepts defined in the calendar ontology, the reasoning system classifies these event instances with respect to their properties – based on the kind of event (is it a business or private event?), its attendees as well as recurrence settings. So in case, the supervisor of the owner attends the meeting, it is therefore categorized as an important event.

4.3 Application Enhancement

Apart from organizing personal information in a taxonomic structure within the T-box, ontology-based reasoning in fact derives entailed knowledge. The entailed information can then be used to enrich applications, which in this case originally provided the data beforehand. McAnt avails this fact in the form of even “smarter” groups.

To achieve this, McAnt recursively retrieves all sub-concepts of the concept *Person* defined in the agent ontology and finally creates appropriate groups in the Address Book. Afterwards the groups are populated according to the classification results obtained from the reasoner. These "intelligent" groups help to navigate through the address book database by classifying the contacts with respect to their relationship with the owner. Unlike the rather limited possibilities of group definitions in the Address Book the ontology-based approach provides the full expressiveness of OWL DL. Latter makes use of the social relationships between the entities defined in the Address Book and therefore provides more sophisticated possibilities for creating smart folders. Furthermore, the logical consistency of the folder definitions is maintained, since conflicting definitions can be detected through the reasoner.

5 Discussion

In this work we presented our initial ideas to support Desktop Applications by rich semantic representations and reasoning mechanisms. In using extensive background knowledge formalized in ontological structures using the expressive OWL language, we were able to achieve knowledge integration across Apple's personal information management components, the Address Book and the iCal application. Furthermore, we highlighted how the functionality of those components can be enhanced by making use of ontology-based formalization and reasoning.

Taking advantage of the full strength of the OWL DL language, the smart groups mechanism implemented in Apple's Address Book application can be tremendously improved. The specification of smart folders in Apple's current implementation is limited to simple sequences of either conjunctions or disjunctions of predefined literals not covering the given social links between entries nor incorporating hierarchical groups. A further drawback of Apple's approach is the lack of any logical reasoning support. Not even simple consistency checks on the represented data are made. Therefore it easily happens that a smart group is given an unsatisfiable description.

The work reported here is very much in line with the idea of a Social Semantic Desktop [5]. A rich modeling of world knowledge (such as the framework of social relations in our agent ontology) is an essential prerequisite for standardizing, linking and wrapping information on the desktop. However, in contrast to other approaches that are based on simple RDF representations, we decided to apply the richer ontology language OWL, originally developed for the Semantic Web, and DL-based logical reasoning.

References

1. Berner-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American **284** (2001) 34–43
2. RDFWeb: The FOAF project (2005) `<http://xmlns.com/foaf/0.1/>`.
3. Horrocks, I., Patel-Schneider, P.: A proposal for an OWL rules language. In: Proceedings of the Thirteenth International World Wide Web Conference (WWW'04), ACM (2004) 723–731
4. Haarslev, V., Möller, R.: Racer system description. In: International Joint Conference on Automated Reasoning (IJCAR'01), June 18–23, 2001, Siena, Italy, Springer (2001)
5. Decker, S., Frank, M.: The Social Semantic Desktop. Technical Report DERI-TR-2004-05-02, Digital Enterprise Research Institute (DERI), Insbruck (2004)