Experience Report: Ontological Reasoning for Context-aware Internet Services *

Alessandra Agostini   Claudio Bettini   Daniele Riboni

DICo, University of Milan
via Comelico 39, I-20135 Milan, Italy
{agostini,bettini,riboni}@dico.unimi.it

Abstract

This paper reports our experience with representing and reasoning with context information within the CARE middleware. CARE was developed to support context-aware service adaptation for mobile users. Expressiveness and computational issues are discussed and the specific solution adopted in CARE is presented.

1 Introduction

The research group of the DaKWE laboratory at the University of Milan has been working for the last three years at the specification and implementation of a middleware – named CARE[1] – to support context-aware service adaptation for mobile users. CARE has three major goals: a) supporting the fusion and reconciliation of context data obtained from distributed sources, b) supporting context dynamics through an efficient form of reasoning, and c) capturing complex context data that go beyond simple attribute-value pairs. While the second goal has been considered in other approaches [9, 16], it becomes more difficult to achieve when different sets of inference rules are provided by distributed sources. Even more difficult is to conciliate efficient reasoning with the expressiveness requirements imposed by the third goal.

The CARE middleware and its underlying technical solutions have been presented elsewhere [1, 5]. In this paper we report our specific experience with the trade-off between expressiveness and efficiency imposed by the above requirements. The solution proposed in CARE will be illustrated based on the experience in developing a specific prototype application.

*This work has been partially supported by Italian MIUR (FIRB "Web-Minds" project N. RBNE01WEJT_005).

[1]Context Aggregation and REasoning middleware.

2 Expressiveness issues

In our framework we need to model both simple context data such as device capabilities or current network bearer, and socio-cultural context information describing, for instance, the user current activity, the set of persons and objects a user can interact with, and the user interests. While the first category, that we call *shallow* context data, can be naturally modeled by means of attribute/value pairs, the second one calls for more sophisticated representation formalisms, such as ontologies and we call it *ontology-based* context data.

2.1 Our experience with OWL-DL

Similarly to other research works (e.g., [6] and [10]), we have adopted OWL [14] as the language for representing ontology-based context data. This choice is motivated by the fact that the description logic languages underlying the *Lite* and *DL* sublanguages of OWL guarantee completeness and decidability, while promising high expressiveness. Moreover, a number of tools already exist for processing OWL ontologies and, being OWL a W3C Recommendation, the available utilities should further increase.

Figure 1 shows part of the OWL-DL ontology we defined for modeling the socio-cultural environment of mobile users, presented in [2]. The ontology is composed by nearly 150 classes and relations that describe features among which there are the user's activities (communications, movements, meetings, ...). On defining this ontology, however, various difficulties arose due to some OWL limitations. Consider for example the *colleague* relation, which is fundamental in modeling the activities performed within an organization. A straightforward definition of the colleagues of an individual A could be: those individuals which are employed by the employer of A. Unfortunately, this definition cannot be expressed in OWL. In fact, the language –for preserving its decidability– does not include the role composition constructors. Similarly, OWL does not include even restricted forms of role-value-maps [3]. A role-

0-7695-2520-2/06 $20.00 © 2006 IEEE

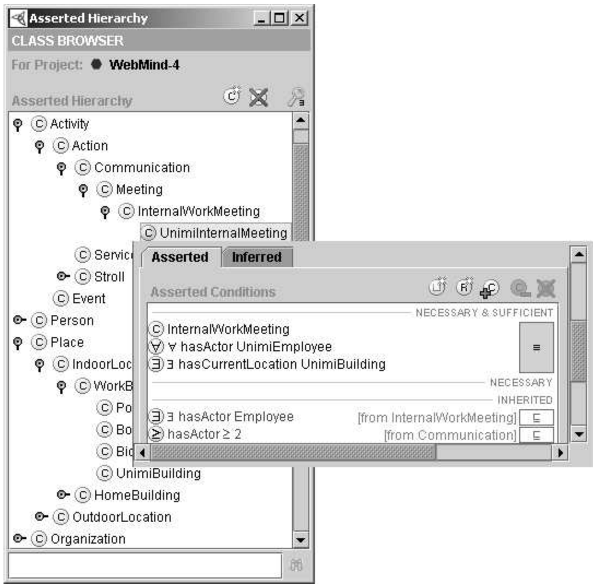

Figure 1. An excerpt of an ontology modeling the socio-cultural context of mobile users

value map $R1 = R2$ defines the class of individuals i such that the set of instances that are connected to i by the relation $R1$ are connected to i also by the relation $R2$. This could be useful, e.g., in defining when an employee is actually in her work location: $Person \sqcap (current\_location = work\_location)$. Due to these expressiveness limitations of OWL, we could not represent certain concepts and had to resort to more ad-hoc and convoluted representations, as shown in Section 4.2.1.

2.2 Issues with proposed OWL extensions

Several proposals have been formulated for overcoming the representational restrictions of OWL. In [12] OWL-DL is mainly considered for consistency checking and semantic interoperability issues, while a totally different modeling formalism is used for actual reasoning. Other approaches have considered augmenting OWL by means of rules. For instance, SWRL [13] extends OWL with Horn clauses. However, the problems underlying the reasoning tasks become undecidable, and the development of optimized tools for reasoning with SWRL subsets is still at an early stage. In [18] a decidable extension of OWL-DL with *DL-safe* rules is proposed. Decidability is preserved by imposing constraints on the specific form of these rules. A reasoner that supports this language –named KAON2– has also been developed. While the majority of DL reasoners currently implement the tableaux calculus, reasoning in KAON2 is based on a transformation of the knowledge base into disjunctive datalog. However, an experimental assessment of the performance of this system is still missing.

In the *semantic eWallet* architecture [7], context reasoning is based on a combination of OWL ontologies and ROWL rules [8]. Besides expressing privacy policies, rules are adopted for modeling relations that cannot be expressed in OWL. Context data (facts) and rules are transformed into Jess statements, and evaluated by the Jess inference engine. However, since the main ambit of the eWallet project is ambient intelligence, performance issues are not as stringent as in providing multi-user Internet services.

Of course, adding rules to OWL is not the only option for augmenting its expressiveness. Unfortunately, the unrestricted introduction in OWL of very expressive description logic constructors such as *role composition*, *feature agreements*, or *role-value maps* would make reasoning undecidable. However, a restricted form of *role-value maps* can be added to the $\mathcal{ALCQI}_{reg}$ language –that is closely related to OWL– without influencing the worst-case complexity of reasoning [3]. The restriction that preserves decidability consists in allowing only *boolean combinations of basic relations* to appear in the role-value map (i.e., role compositions are not allowed). Since the well-known *Racer* reasoner supports the closely related logic $\mathcal{ALCQHI}_{R^+}$ (usually referred as $\mathcal{SHIQ}$), this extension is particularly interesting. Clearly, this restricted form of role-value maps does not overcome all the main weaknesses of OWL. As a matter of fact, role-value maps can only define concepts, and thus they cannot express relations such as *colleague* in the form we illustrated in Section 2.

To this respect, a decidable extension of $\mathcal{SHIQ}$ with *acyclic role inclusion axioms* of the form $R \circ S \sqsubseteq R$ or $R \circ S \sqsubseteq S$ is discussed in [15]. This constructor is useful for expressing the propagation of one property along another property. As an example, the propagation of the $current\_location$ property along the is_part_of property is expressed by $current\_location \circ is\_part\_of \sqsubseteq current\_location$. The DL reasoner *FaCT* provides support for this constructor, which is particularly desirable for reasoning with spatial properties.

A further solution for augmenting the expressivity of OWL is to support *concrete domains*. A concrete domain is essentially a datatype with a set of associated predicates. Including concrete domains into a description logic framework would allow the implementation of a hybrid reasoning mechanism, composed by DL reasoning with the abstract domain, and by external reasoning with the concrete domain predicates. Concrete domains could be particularly useful for modeling and reasoning with spatio-temporal information, which is crucial for representing the user's context. In the last years various description logic languages including concrete domains have been proposed (e.g., [11]).

Currently, the support of concrete domains in OWL is rather weak.

3 Computational issues

Even if OWL-Lite and OWL-DL guarantee completeness and decidability, performing reasoning tasks with an OWL ontology could be computationally unfeasible, especially when providing an interactive service to a possibly huge number of users. Despite several assessments on the performance of reasoning with description logics are available, we performed some tests in order to assess the feasibility of executing ontological reasoning at the time of the service request with our specific OWL-DL ontologies. In the prototype application presented in Section 4, the reasoning task essentially consists in querying the ontology for the membership of instances to specific classes (e.g. representing the specific user activities). All queries have been evaluated using the *Racer* reasoner on a two-processor Xeon 2.4 GHz workstation with 1.5GB of RAM. Queries from $Q1$ to $Q6$ are made on different classes (e.g., *ContactProfile*, *WorkMeeting*) having an increasing number of instances. Results are shown in Figure 2, in logarithmic scale. As expected, the query response time is strongly correlated to the number of instances of the examined ontology class as well as to the depth of the class within the ontology hierarchy. In particular, the query response time dramatically increases whenever the number of instances of a class is more than 250. Our results confirmed that the execution of ontological reasoning at the time of the service request is unfeasible, even having a small ontology populated with few instances. Moreover, the introduced delay on servers handling a large number of user requests can seriously affect the scalability of the system. This consideration led us to the architectural choices that are presented in Section 4.

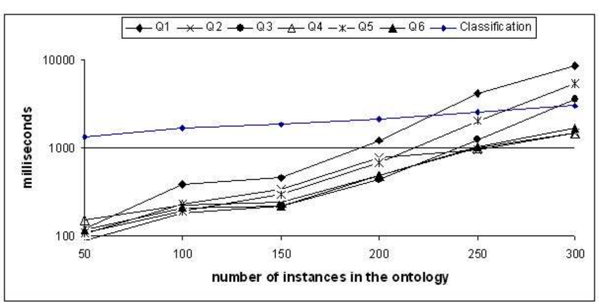

Figure 2. Performance results about ontological reasoning with socio-cultural information.

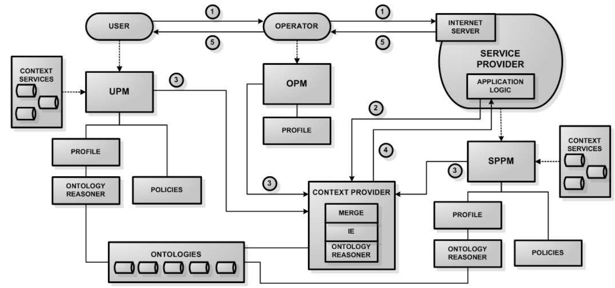

Figure 3. The *CARE* middleware architecture.

4 The *CARE* middleware hybrid approach

In this section we recap the main features of the *CARE* middleware (see Figure 3). We illustrate our choices by means of a prototype application taking advantage of *CARE*.

4.1 Architecture overview

In our framework the contextual information, being by nature distributed, is managed by different entities, namely: the user with her devices; the network operator with its infrastructure; and the service provider. We call *profile* a subset of context information collected and managed by a certain entity. Profiles are represented by using CC/PP [17]. Profiles include both shallow context data and ontology-based context data which are expressed by means of references to ontological classes and relations inserted in the CC/PP profile. Each entity has a dedicated Profile Manager for handling its own context data: the User Profile Manager (UPM); the Operator Profile Manager (OPM); and the Service Provider Profile Manager (SPPM). Both the user and the service provider can declare policies in the form of rules over profile data which guide the adaptation and final personalization of the service. The CONTEXT PROVIDER module is in charge of building the aggregated context information for the application logic. In particular, it evaluates adaptation policies and solves possible conflicts arising among context data and/or policies provided by different entities. The ad-hoc rule-based reasoning services of the CONTEXT PROVIDER are particularly efficient if no ontological reasoning is performed, having linear complexity [5]. Experimental results have shown that the evaluation of rules is executed in few milliseconds.

4.2 Introducing ontological reasoning

We extended the original *CARE* architecture by introducing ONTOLOGY REASONERS with the main goal of deriving

Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)
0-7695-2520-2/06 $20.00 © 2006 IEEE

new ontology-based context data based on the data explicitly available. For a framework in which efficiency is a fundamental requirement, the introduction of ontological reasoning has been particularly challenging. The experimental results briefly reported in Section 3 imposed the choice of an off-line form of ontological reasoning, i.e., anticipating, whenever it is possible, ontological reasoning before a user requests a service.

4.2.1 An ontology for modeling user's activities

The user's current activity should be taken into account for properly adapting the service in terms of both the displayed content and of the most suitable access modalities. As an example, consider the case of a user of *POIsmart* [4], a prototype system for the recommendation and management of an extended form of points of interest. POIsmarts can be considered as the convergence between virtual points of interest (Web bookmarks) and physical points of interest (GPS bookmarks); Figure 4 shows some screenshots of the client application. In order to properly adapt the service, both the service content (e.g., the displayed POIsmarts) and the most suitable modalities are chosen considering the user's current activity. For instance, if the user is strolling for leisure, she would be probably interested in amusing spots like shopping centers and pubs, and she would like to receive rich multimedia content. On the other hand, a user who is involved in a work meeting would be probably interested in less frivolous items, such as Web resources related to the project she is working on.

In order to implement the above and similar scenarios, we have defined an OWL-DL ontology for modeling the activities of mobile users. In particular, among the user's activities, we would like to identify when an employee is involved in a work meeting. Moreover, mainly for handling privacy issues, we want to distinguish the meetings among colleagues (*'InternalWorkMeeting'*) from the meetings involving people external to the organization (e.g., the customer). A possible definition of *InternalWorkMeeting* is *an activity involving at least two colleagues being in the same place located in their organization buildings*. We have outlined the main expressiveness weaknesses of OWL in Section 2. In this case, the lack of constructors for defining the *colleague* relation brought us to define the *InternalWorkMeeting* concept as the union of the possible instances of internal work meetings of specific organizations:

$$\mathit{InternalWorkMeeting} \equiv \mathit{UnimiInternalMeeting} \sqcup \mathit{PolimiInternalMeeting} \sqcup \ldots$$

Obviously, this definition is not general, but can only fulfill the requirements of a well-defined group of users. The internal work meeting of a specific organization (in this example, *Unimi*) is defined as an activity performed in a place

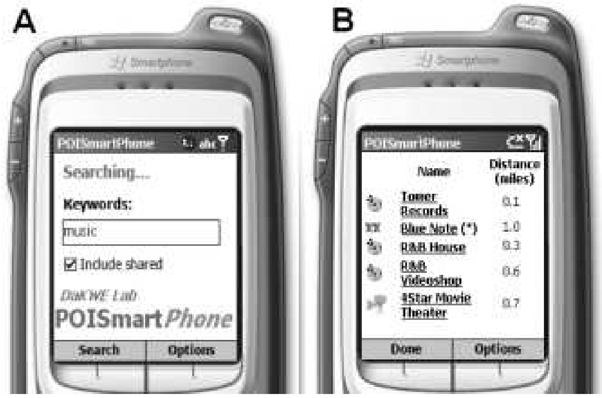

Figure 4. Screenshots of the *POIsmart* service

belonging to that organization by at least two people which are employees of the organization:

$$\mathit{UnimiInternalMeeting} \equiv \mathit{Activity} \sqcap \geq 2\,\mathit{Actor} \sqcap \forall \mathit{Actor.UnimiEmployee} \sqcap \exists \mathit{Location.UnimiLocation}$$

$$\mathit{UnimiEmployee} \equiv \mathit{Employee} \sqcap \exists \mathit{Employer}.\{\mathit{unimi}\}$$

4.2.2 Hybrid reasoning

The hybrid approach implemented in *CARE* is based on a loose interaction between ontological and rule-based reasoning. In order to illustrate the hybrid mechanism, suppose that a POIsmart user declared a policy rule asking to give priority to POIsmarts regarding work when involved in an internal meeting:

If *Activity = 'InternalWorkMeeting'*
then *InterestCategories.add (Work)* (1)

Since the rule precondition predicate *Activity* is an ontology-based context parameter, its value must be inferred through ontological reasoning before evaluating the rule. While rule-based reasoning is performed at the time of the service request, ontological reasoning is mostly performed asynchronously by profile managers. *Off-line ontological reasoning* is local to a single profile manager (the UPM and SPPM in the current implementation). It is performed before the user requests a service, and it is fired by local activation rules (e.g., change on a profile attribute). In this case, since the definition of *InternalWorkMeeting* regards the user's location, which is known by the UPM, ontological reasoning is performed by the UPM, and the corresponding activation rule is:

If *changes(currentLocation)* **then** *execOntReasoning*.

In this way, when the user requests a service, the UPM profile will already contain the right value for the *Activity* attribute (i.e., *InternalWorkMeeting*) and rule (1) will fire.

Ontological reasoning is performed at the time of the service request only in particular cases by the CONTEXT PROVIDER module. It is performed when some crucial ontology-based attributes have no values and these values can be possibly obtained after populating a shared ontology with the integrated profile. It is the duty of the service provider, on the basis of the specific supplied service, to carefully decide which attributes are crucial. In this case, after ontological reasoning, policies are re-evaluated, determining the aggregated profile. Actually, analyzing pragmatically various case studies, we believe that there are only few cases in which ontology-based data cannot be precomputed.

Conclusions

Our experience on the development of a middleware for context-aware Internet services lead us to the conclusion that, despite the expressive power of current ontology languages is required – and sometimes it is not sufficient – to express high-level context data, ontological reasoning is computationally still very expensive and for this reason should be avoided at the time of service request.

References

[1] A. Agostini, C. Bettini, N. Cesa-Bianchi, D. Maggiorini, D. Riboni, M. Ruberl, C. Sala, and D. Vitali. Towards Highly Adaptive Services for Mobile Computing. In *Proceedings of IFIP TC8 Working Conference on Mobile Information Systems (MOBIS)*, pages 121–134. Springer, 2004.

[2] A. Agostini, C. Bettini, and D. Riboni. Loosely Coupling Ontological Reasoning with an Efficient Middleware for Context-awareness. In *Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous 2005)*, pages 175–182. IEEE Computer Society, 2005.

[3] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[4] C. Bettini, N. Cesa-Bianchi, and D. Riboni. A Distributed Architecture for Management and Retrieval of Extended Points of Interest. In *Proceedings of the first International Workshop on Services and Infrastructure for the Ubiquitous and Mobile Internet (SIUMI'05), To appear*. IEEE Computer Society, 2005.

[5] C. Bettini and D. Riboni. Profile Aggregation and Policy Evaluation for Adaptive Internet Services. In *Proceedings of The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (Mobiquitous)*, pages 290–298. IEEE Computer Society, 2004.

[6] H. Chen, F. Perich, T. W. Finin, and A. Joshi. SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications. In *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous 2004), Networking and Services*, pages 258–267. IEEE Computer Society, 2004.

[7] F. Gandon and N. M. Sadeh. A Semantic E-Wallet to Reconcile Privacy and Context Awareness. In *Proceedings of ISWC 2003, Second International Semantic Web Conference*, pages 385–401. Springer, 2003.

[8] F. L. Gandon, M. Sheshagiri, and N. M. Sadeh. ROWL: Rule Language in OWL and Translation Engine for Jess. Technical report, Carnegie Mellon University, February 2004.

[9] B. Grosof. Prioritized Conflict Handling for Logic Programs. In *Proceedings of the International Logic Programming Symposium (ILPS)*, pages 197–211, 1997.

[10] T. Gu, X. H. Wang, H. K. Pung, and D. Q. Zhang. An Ontology-based Context Model in Intelligent Environments. In *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2004.

[11] V. Haarslev, C. Lutz, and R. Möller. A Description Logic with Concrete Domains and a Role-forming Predicate Operator. *Journal of Logic and Computation*, 9(3):351–384, 1999.

[12] K. Henricksen, S. Livingstone, and J. Indulska. Towards a hybrid approach to context modelling, reasoning and interoperation. In *Proceedings of the First International Workshop on Advanced Context Modelling, Reasoning And Management, UbiComp 2004*, 2004.

[13] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission, W3C, May 2004.

[14] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. From SHIQ and RDF to OWL: The Making of a Web Ontology Language. *Journal of Web Semantics*, 1(1):7–26, 2003.

[15] I. Horrocks and U. Sattler. Decidability of SHIQ with complex role inclusion axioms. *Artificial Intelligence*, 160(1-2):79–104, 2004.

[16] R. Hull, B. Kumar, D. Lieuwen, P. Patel-Schneider, A. Sahuguet, S. Varadarajan, and A. Vyas. Enabling Context-Aware and privacy-Conscius User Data Sharing. In *Proceedings of the 2004 IEEE International Conference on Mobile Data Management*, pages 187–198. IEEE Computer Society, 2004.

[17] G. Klyne, F. Reynolds, C. Woodrow, H. Ohto, J. Hjelm, M. H. Butler, and L. Tran. Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation, W3C, January 2004. http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/.

[18] B. Motik, U. Sattler, and R. Studer. Query Answering for OWL-DL with Rules. In *Proceedings of The Semantic Web ISWC 2004: Third International Semantic Web Conference*, pages 549–563. Springer, 2004.

0-7695-2520-2/06 $20.00 © 2006 IEEE