Situation-aware Mobile Service Recommendation with Fuzzy Logic and Semantic Web

Alessandro Ciaramella

IMT Lucca Institute
for Advanced Studies,
Piazza San Ponziano 6,
55100 Lucca (Italy)
a.ciaramella@imtlucca.it

Mario G.C.A. Cimino, Beatrice Lazzerini, Francesco Marcelloni

Dipartimento di Ingegneria dell'Informazione: Elettronica, Informatica,
Telecomunicazioni,
University of Pisa, Via Diotisalvi 2,
56122 Pisa (Italy)
{m.cimino, b.lazzerini, f.marcelloni}@iet.unipi.it

Abstract — **Today's mobile Internet service portals offer thousands of services and mobile devices can host plenty of applications, documents and web URLs. Hence, for average mobile users there is an increasing cognitive burden in finding the most appropriate service among the many available. On the other hand, methodologies such as bookmarks and resource tagging require a great arranging effort to handle increasing resources. To help mobile users in managing and using this personal information space, new levels of granularity should be introduced in the organization of services, together with some degree of self-awareness. This paper proposes a situation-aware service recommender that helps locating services proactively. In the recommender, a semantic layer determines one or more user current situations by using domain knowledge expressed in terms of ontology and semantic rules. A fuzzy inference layer manages the vagueness of some contextual condition of these rules and outputs an uncertainty degree for each situation. Based on this degree, the recommender proposes a set of specific resources.**

Keywords - Context-awareness; Fuzzy Inference System; Mobile Service Recommender; Web Ontology; Semantic Rules

I. INTRODUCTION

Mobile Internet is growing incredibly. The most important mobile service marketplaces host thousands of applications, ranging from entertainment (games, songs, etc.) to information (weather forecast, traffic maps, news, etc.), from transactions (money transfer, airline reservation, etc.) to productivity (notepad, voice reader, etc.). Applications can be used in a number of circumstances and for different purposes. Hence, it is very difficult to categorize services only on the basis of their functions. Mobile services may belong to several categories depending on the use case and the customer's individual preferences [1].

The lack of a standardized classification makes the retrieval of a service very difficult. Searching and browsing are the most used mechanisms to locate services in repositories. Usually, finding the best-fitting service is a time-expensive task, and service marketplaces are so large that users cannot learn about all of the available components. Furthermore, the user interface on a mobile terminal is less comfortable and less convenient to control with respect to a personal computer. Last but not least, once installed, applications need to be configured and launched with a set of proper parameters, which often vary in dependence of specific user circumstances [2].

Many applications can be stored into a today's smart phone, as well as a number of documents and links to external resources. The personal information space of a mobile user is then growing, in dimensionality and variety, more than the interaction capabilities of the devices. Indeed, mobile devices have to be small to be portable, and then they feature small screens and keypads. As a consequence, a significant cognitive effort is required to users in mobility to find and configure the most appropriate services or resources among the many available [3][4].

In this paper, we propose a context-aware service recommender for mobile devices, which allows locating resources by taking the user's situation into account [3][4]. More specifically, the proposed approach enables the delivery of recommendations in a proactive way. This is achieved by the integration of a fuzzy inference layer and a semantic layer. The semantic layer infers one or more current situations of the user by exploiting domain ontology and semantic rules. The fuzzy inference layer handles the vagueness of some conditions of these rules and outputs an uncertainty degree for each situation determined by the semantic layer. Based on these uncertainty degrees, a situation is recognized. This situation allows the identification of specific tasks, on the basis of domain knowledge expressed in terms of task-based ontology and rules [5][6]. Finally, the specific current task together with contextual information is used to recommend a set of resources, referred by means of a Label (or Tag)-based file system [7].

II. BACKGROUND

To organize personal resources independently of their location, a bookmark management system is usually employed. The function of bookmarks is to offer an

associative memory for personal usage. Conventional bookmarks contain a URL (or local path) and a title of the resource [8]. Bookmarks reduce the cognitive and physical loads of managing URL addresses, and facilitate the return to groups of related resources. Users collect bookmarks to create their own personal information space and share it with others [9]. However, organizing bookmarks is labor-intensive, requires a lot of time, and is difficult to do. In fact, typically users do not organize bookmarks [8]. Web usability studies show that bookmark lists are far from representing an effective personal information space [9].

A number of researches have been performed to enhance bookmark functionality. In [11], context-dependent bookmarks have been discussed, with a method based on the automatic extraction of representative keywords of resources. The automatic extraction of representative keywords is applicable only to documents. In the case of applications, descriptors should convey the user intention, which is difficult to extract without a semantic layer representing high-level concepts.

Recently, the *tagging* paradigm for information organization has become popular, especially in the context of collaborative tagging systems for managing shared bookmarks or public digital images. Resources are tagged by annotating them with simple descriptors. Conjunctions of tags can be used to narrow down the search space and, at the limit, to identify a limited set of resources such as a folder path. Information organization based on tags is capable of overcoming many problems inherent in hierarchical file systems [12][13]. The information about tags can also be represented in an ontology, with the advantage that extensions of the data model and integration with other semantic-aware applications are easy to realize.

Tags are likely to quickly grow considering a great collection of resources. Hence, information represented by tags cannot be efficiently exploited without a proper user interface (using a laptop or desktop device), or without a further level of semantics, which helps the system take the current intention of the users into account. Key to support mobile users with an efficient access to resources is an intelligent platform that mediates between services and users by observing the user activity.

The research about providing personalized services has been carried out by many researchers in recommendation systems [14]. In the literature of mobile computing, the use of context information is introduced in terms of *implicit input* from changes in the environment [15]. This model is usually referred to as *context-aware*, because the output of the system depends on who is using the application, where, when, and in which situation. Designing context-aware applications involves two main steps: (i) designing a set of rules to infer high-level situations and (ii) designing proper input drivers to gather context information from the environment. Several projects consider the use of ontology as a key technology for context-awareness. In the framework of semantic web, an ontology is a knowledge model that describes a domain of interest using semantic aspects and structure. An ontology consists of: (i) facts representing explicit knowledge, consisting of concepts, their properties,

and instances that represent entities described by concepts; (ii) axioms and predicates representing implicit knowledge, by means of rules used to add semantics and to derive knowledge from facts [16]. In [17], a comparative analysis shows that the most promising assets for context modeling in ubiquitous computing environments can be found in the use of ontology.

To reflect the varying nature of context and to ensure a universal applicability of context-aware systems, context is typically represented at different levels of abstractions [3]. At the first level of raw context sources there are context data coming from sensor devices, or user application. At this level, logic embodied in semantic languages does not allow a treatment of uncertainty and imprecision existent in real world [18]. For instance, a typical smart phone GPS receiver provides a device position with dynamic accuracy ranging from some meters to hundreds of meters, depending on many environmental variables. Again, for instance, the time and location provided by the user's calendar can be considered in practice only as ideal references, because real events usually happen *approximately* at the referred time and place. Hence, the recognition of situation from the environment should rely on a vague characterization. Furthermore, these situations are often connected to specific user needs, and then the system should offer a specification mechanism that is intuitive, for instance in terms of standardized natural language, as guaranteed by employing fuzzy linguistic terms [19].

III. OVERALL ARCHITECTURE

In our implementation, the situation-aware service recommender is running on the mobile device as an advanced menu, whose elements are dynamically updated, according to the different situations in which the user is involved. The overall system architecture is shown in Fig. 1. In the server side, there are two main modules, i.e., the *fuzzy engine* and the *semantic engine*. The fuzzy engine module is in charge of assessing conditions that are inherently vague, such as mobility and proximity state of users. For instance, "the user is close to a place".

The domain model and the behavior of the system are instead handled in the semantic engine module, which infers the current situation of the users and suggests the most useful resources for that situation. The control flow of the application is steered by the *application controller* module, which manages the activities of each module, granting access to different functions and data sources. The *contextual data sources* package comprises a set of interfacing modules for different data sources, such as geographical maps, users' calendars and positions. In particular, numerical data concerning user positions are fed by the *location detector* module. This module provides outdoor/indoor location estimation, also on the basis of several possible technologies, such as GPS, GSM, WiFi [23]. Regardless of the available technologies, the location detector provides a generalized interface in terms of position and accuracy.

On the client side, the *label-based resource access* [13] module is supplied by the application controller module with a set of labels and contextual parameters. This information is
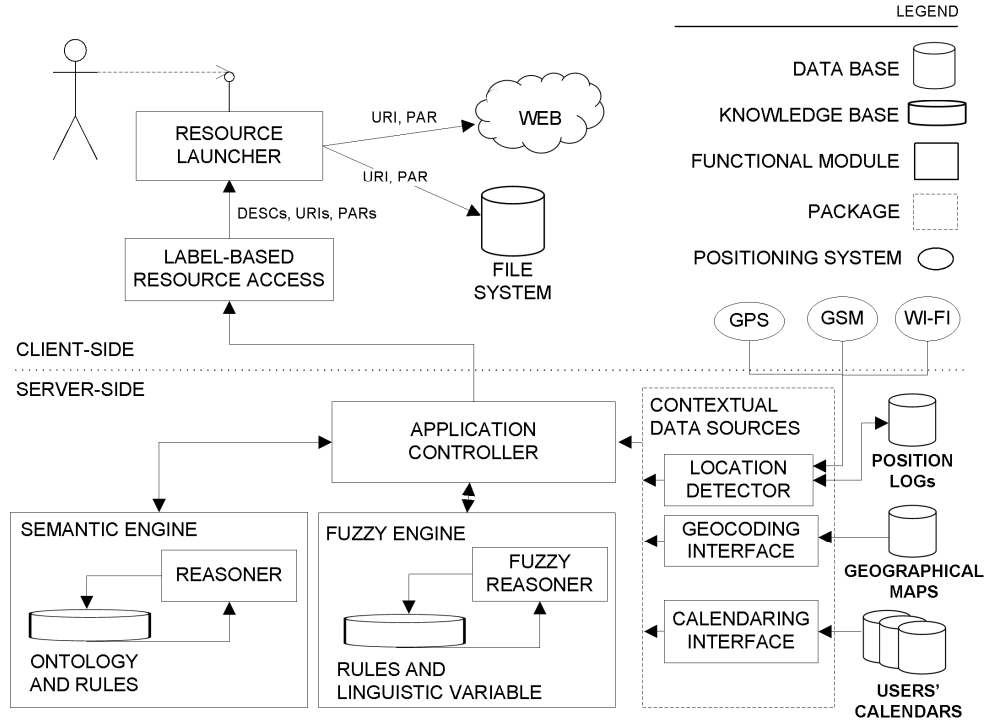
Figure 1. Overall architecture of the context-aware service-recommender.

used to locate and adapt recommended resources. Finally, the selected resource is identified and can be started with the *resource launcher* module. In the following, the paper is focused on the design of the semantic and the fuzzy engine modules, considering also an evaluation case.

IV. THE SEMANTIC ENGINE MODULE

To recommend services inherent in the current user task, the system takes the current user situation into account. According to [16] the term "situation" is a business level concept that allows targeting precisely and at different levels of granularity the demand of the user at a certain time. In our system, each situation is devoted to identify a collection of user tasks. In a task-navigation paradigm [3], the user is supported to find appropriate services by relying on a task ontology, which represents common sense knowledge about his usual activities. In order to suggest in a proactive manner tasks and services actively, i.e., without the need for initial user input, the user context is a fundamental vehicle. Context refers to any relevant information that can be used to characterize a user [22]. Therefore, a situation can be modeled as a collection of context information that is invariant as long as the situation occurs [16]. For instance, the situation "meeting" can be inferred from a set of context information such as "user is stationary", "user is located in the scheduled place at the scheduled time", "user is close to the meeting organizer", and so on.

Another important advantage of using context-information is the possibility of deriving contextual parameters to adapt the identified service to the current demand of the user. Hence, the full goal of the ontology is to identify a set of service descriptors together with a set of contextual parameters. Furthermore, to make the ontology independent of specific applications and related path installations, and of specific number, type and sequence of parameters, two abstraction mechanisms have been introduced in the system, by means of the following respective modules: the *label-based resource access,* which allows the exact localization of an application or a document, described more generically as a service in the ontology, and the *resource launcher,* which enables the forwarding of the gathered parameters, and the launching of the selected application.

The core of the semantic engine module is represented by the task-based service recommendation ontology and the related semantic rules. The ontology has been developed using the Web Ontology Language (OWL [5]), a W3C standard well-supported in most semantic engines. In the upper context-based ontology general context information is represented by basic concepts such as *User, Calendar, Device, Time,* and *Place.* In Fig. 2, concepts have been enclosed in oval shapes.

In particular, general concepts such as *Time* and *Place* are inherited from publicly available ontologies [20][21], according to the best practices of reusing domain ontologies. In the figure, external ontologies are enclosed in dashed rectangular shapes. Concepts are connected by properties, represented with directed edges in the figure.

The model comprises a set of rules to infer the situation on the basis of the context-based ontology. Rules are expressed in the Semantic Web Rule Language (SWRL [6]), an emerging standard that extends OWL with additional rule-based knowledge representation. In terms of expressiveness, this reasoning standard corresponds to

description logics, a particular decidable fragment of first order logic, and it is named OWL DL [5]. Fig. 3 shows an example of rule in human readable syntax (a), commonly used in the literature, and in natural language (b). We point out that there are two types of antecedent conditions, i.e., crisp (binary) and fuzzy, represented in Fig. 3 in bold and italic bold, respectively. The condition "is a participant" is derived from the user's calendar, and is inherently crisp, whereas the other conditions can be assessed only with vagueness. This implies that also the conclusion inferred from the rule is characterized by vagueness. Although web ontology is the most promising assets for context modeling for ubiquitous computing [17], the semantic web formalisms do not allow the representation of uncertainty [24]. Thus, in this paper we propose a design methodology in which the semantic engine does not handle directly the uncertainty.
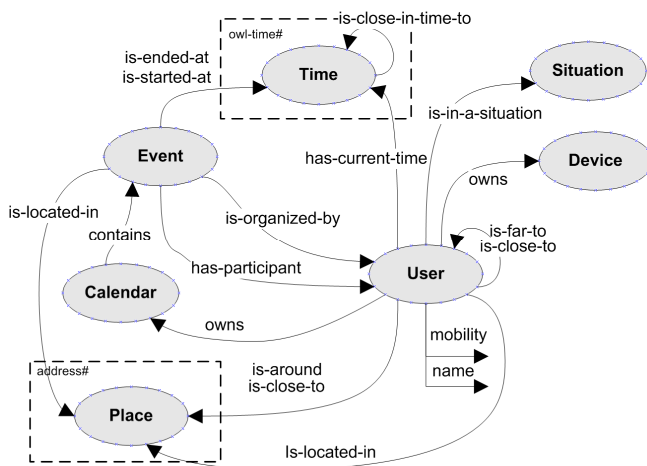

Figure 2. The upper context-based ontology.

```
  owns(?user1, ?aCalendar)
^ contains(?aCalendar, ?anEvent)
^ is-located-in(?anEvent, ?aPlace)
^ is-started-at(?anEvent, ?anEventTime)
^ has-current-time(?user1, ?user1Time)
^ has-participant(?anEvent, ?user1)
^ has-participant(?anEvent, ?user2)
^ is-close-in-time-to(?user1Time, ?anEventTime)
^ is-close-to(?user1, ?aPlace)
^ is-far-from(?user1, ?user2)
^ PreMeeting(?aSituation)
⟹ is-in-a-situation(?user1, ?aSituation)
```
(a)

```
IF user1 IS A PARTICIPANT to the scheduled event
AND user2 IS A PARTICIPANT to the scheduled event
AND user1 IS CLOSE TO the scheduled place
AND user1Time IS CLOSE IN TIME TO the scheduled time
AND user1 IS FAR FROM user2
THEN user1 IS IN A SITUATION of pre-meeting
```
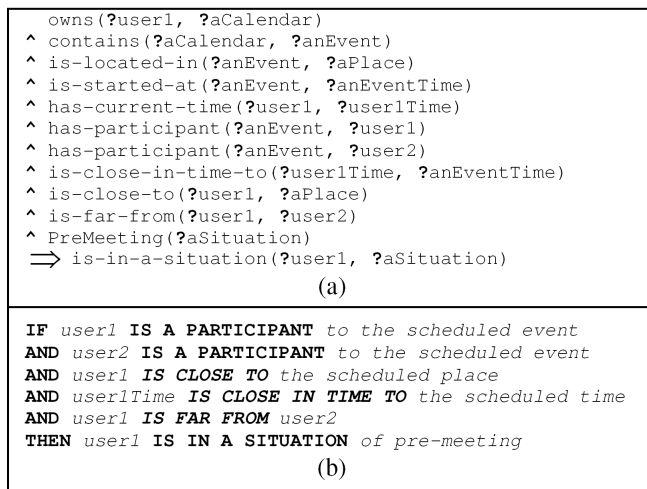(b)

Figure 3. An SWRL rule to infer pre-meeting situation. (a) human readable syntax, (b) natural language.

More specifically, when the semantic rules are characterized by fuzzy conditions, the application controller asks the fuzzy engine for their evaluation. The fuzzy engine

returns a certainty value in [0, 1] for each uncertain condition. If the certainty value is larger than zero, the condition is considered to be true in the semantic inference. Otherwise the condition is considered to be false. When the semantic engine infers a situation, the fuzzy engine, based on these fuzzy conditions, computes a certainty degree for this situation.

Once a situation is inferred (with a certainty degree larger than zero), a task-based ontology allows connecting a situation to specific tasks, and then to specific services to be recommended. In Fig. 4 the upper task-based ontology is represented. In particular, a dashed edge has been used to represent a property that is not implemented in the ontology, but is conceived only for a better understanding (i.e., *requires*).
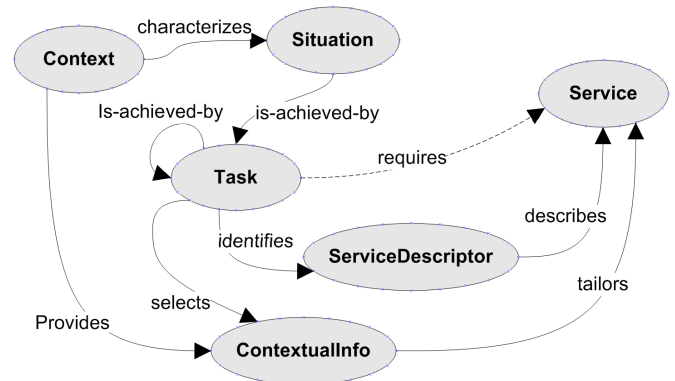

Figure 4. The upper task-based ontology.

V. THE FUZZY ENGINE MODULE

There is some uncertainty in many contextual conditions related to real-world events. For instance, the condition "*user1* is close to *the scheduled place*", in Fig. 3.b, can be assessed only with a degree of uncertainty. This uncertainty can arise from lack of exact information in the user's calendar, or from noisy signals coming from the location sensors. Fuzzy set theory and fuzzy logic are particularly suited for dealing with such uncertainty.

The fuzzy engine module comprises a set of abstract fuzzy clauses. To ensure reusability, these clauses are designed in an abstract form, i.e., independent of locations, events, timetables, users, etc.

As an example, let us consider the rule shown in Fig. 3.b. Here, the clauses "*user1* is close in time to *the scheduled time* and *user1* is close to *the scheduled place* and *user1* is far from *user2*" are fuzzy conditions related to temporal and spatial proximity. To fix ideas, Fig. 5 shows an example of GPS track, provided by a smart phone. A user moves from Q to P to participate to a meeting event.

In the fuzzy engine, spatial and temporal proximities are expressed as linguistic variables Δs and Δt, respectively. The number and meaning of the possible linguistic values for these variables are application-dependent. In our case study, we partitioned the universe of definition of these

variables with trapezoidal membership functions, appropriately extracted from experimental data. We adopted two (*low* and *high*) and three (*low*, *medium* and *high*) linguistic values for Δs and Δt, respectively.

Figure 5. An example of GPS track, provided by a user smart phone.

In particular, let $(\mathbf{s}, t)$ be the reference location and time for the event scheduled in the user's calendar. Let $(\mathbf{s}_1, t_1)$ and $(\mathbf{s}_2, t_2)$ be the current location and time of *user1* and *user2*, respectively, provided by their mobile devices. Let $\Delta t_1 = |t_1 - t|$, $\Delta s_1 = \|\mathbf{s}_1 - \mathbf{s}\|$ and $\Delta s_{12} = \|\mathbf{s}_1 - \mathbf{s}_2\|$ be the current user temporal and spatial proximities. Hence, the fuzzy rule correspondent to the semantic rules in Fig. 3 is

```
IF  Δt₁  is LOW and
    Δs₁  is LOW and
    Δs₁₂  is HIGH
THEN situation is Pre-Meeting
```

This rule is generated dynamically once the semantic engine has recognized the *pre-meeting* situation. To this aim, the certainty degrees of the conditions are returned to the application controller. For each condition with a certainty degree larger than zero, the application controller inserts the corresponding property in the ontology and triggers the semantic engine. Hence, the semantic engine can infer one or more situations. In the fuzzy engine, we adopted trapezoidal membership functions, and implemented the logical and the implication operators by minimum. The use of trapezoidal membership functions helps constrain the number of activated conditions, thus limiting the number of concurrently inferred situations.

The application controller associates the corresponding certainty degree with each situation. The certainty degree of a situation is important to consider the order with which services are recommended. If more than one situation is recognized, all the related services are recommended, with an order depending on the certainty degree.

VI. EVALUATION CASE STUDY

We applied the situation-aware mobile service recommender to a real business case, assessing the effectiveness of the semantic engine and fuzzy engine modules. The evaluation case study concerns a pharmaceutical consultant in typical business situations, in which an *Apple iPhone 2G* smart phone is employed. Since the first interviews, we realized that a pharmaceutical consultant would better benefit from a new generation smart phone, to gain a more effective way of managing messages, of accessing contact data, of opening documents, and of communicating with clients and colleagues during meetings or while traveling. The most common use case is inherent to meetings with medical specialists. Initially, a specific domain model has been added to the upper domain, by means of a series of interviews. In particular, the situations of interest are: (i) *Meeting-Planning*, (ii) *Pre-Meeting*, (iii) *Ongoing-Meeting*, (iv) *Post-Meeting*, (v) *Hospital-Conference,* (vi) *Call-for-Tenders*. For each situation, a set of possible tasks has been defined. For each task, a set of related resources have been characterized, in terms of labels and parameters. This modular methodology guarantees that the resulting domain model is highly reusable. It is worth noting that, once the user has completed the interviews, his demand of mobile services has appreciably increased as shown in Table I.

TABLE I. NUMBER OF DEMANDED SERVICES PER SITUATION, BEFORE AND AFTER THE DEVELOPMENT OF THE TASK-BASED ONTOLOGY.

situation id	before interviews	after interviews
i	6	16
ii	5	15
iii	2	8
iv	2	7
v	2	9
vi	2	8

In the use case, the linguistic variables of the fuzzy engine module have been also tuned. For this purpose, a data set consisting of 50 different training tracks have been employed. These tracks have been generated with different conditions, such as moving by foot, bicycle and car without/with traffic jam, indoor and outdoor events, formal and informal events, periodic and unique events, etc. We experienced that, once tuned, fuzzy variables can be reused for a number of different purposes. We are also experiencing the use of linguistic hedges to adapt this fuzzy inference system to different user needs.

After the tuning process, the system has been tested by considering the business events concerning one overall working week of 2 different pharmaceutical consultants. We managed 53 different test events. During the experimentation, for each event, we simulated different conditions. In particular, by delaying or anticipating the beginning of an event with respect to the time stored in the user's calendar, we tested how the system was reliable to determine the time at which actually each situation occurred.

The system has been able to recognize the right situations related to all the test events under the different conditions. Further, the differences between the time at which actually the situation occurred and the time at which the system recognized the situation were of the order of few seconds. This proves the usefulness of our service recommender which exploits data collected by different sensors to determine the situations of interest with respect to

a recommender based only on the scheduled events stored in the user's calendar.

VII. CONCLUSIONS

In this paper, a situation-aware mobile service recommender is proposed. The study focuses on two important modules, i.e., the fuzzy engine, which takes the assessment into account of real-world inaccurate information, and the semantic engine, which contains the service recommendation ontology and the related semantic rules. Finally, a real evaluation case study is presented, to give a concrete view of the system.

REFERENCES

[1] K. Heinonen and M. Pura, "Classifying mobile services", Proc. Helsinki Mobility Roundtable, Working Papers on Information Systems, Sprouts, Vol. 6, Art. 42, 2006.

[2] S. Figge, "Situation-dependent services — a challenge for mobile network operators", Journal of Business Research, Elsevier, Vol. 57, Issue 12, 2004, pp 1416-1422.

[3] M. Luther, Y. Fukazawa, M. Wagner, A. Kurakake, "Situational reasoning for task-oriented mobile service recommendation", The Knowledge Engineering Review, Cambridge University Press, Vol. 23, 2008, pp. 7-19.

[4] L.W. Goix, M. Valla, L. Cerami, and P. Falcarin, "Situation inference for mobile users: a rule based approach", Proc. International Conference on Mobile Data Management (MDM '07), IEEE Computer Society, 2007, pp. 299-303.

[5] W3C, OWL Web Ontology Language Reference, W3C Recommendation, 10 February 2004, electronically available at http://www.w3.org/TR/owl-ref/.

[6] W3C, SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission 21 May 2004, electronically available at http://www.w3.org/Submission/SWRL/.

[7] Y.L. Tai, S.R. Tsai, G.H. Huang, C.M. Lee, L.J. Tsai, K.F. Ssu, and S.J. Wey, "A label-based file system", Journal of Computers, Vol. 19, No. 4, January 2009.

[8] D. Abrams, R. Baecker, and M. Chignell, "Information archiving with bookmarks: personal web space construction and organization", Proc. ACM SIGCHI conference on Human factors in computing systems (CHI '98), ACM Press, 1998, pp. 41-48.

[9] R. Kanawati and M. Malek, "Cowing: a collaborative

[10] bookmark management system", Proc. International Workshop on Cooperative Information Agents (CIA '02), Springer-Verlag, *Lecture Notes in Artificial Intelligence* 2828, 2001, pp. 34-39.

[11] S. Nakajima, S. Oyama, K. Sumiya, K. Tanaka, "Context-dependent web bookmarks and their usage as queries", Proc. International Conference on Web Information Systems Engineering (WISE '02), Systems Engineering, 2002, pp. 333-341.

[12] P. K. Vatturi, W. Geyer, C. Dugan, M. Muller, and B. Brownholtz, "Tag-based filtering for personalized bookmark recommendations", Proc. 17th ACM Conference on information and Knowledge Management (CIKM '08), ACM, 2008, pp. 1395-1396.

[13] S. Bloehdorn, O. Görlitz, S. Schenk, and M. Völkel, "TagFS, tag semantics for hierarchical file systems", Proc. 6th International Conference on Knowledge Management, (I-KNOW'06), 2006.

[14] J. Hong, E.-H. Suh, J. Kim, and S.Y. Kim, "Context-aware system for proactive personalized service based on context history", Expert Systems with Applications, Elsevier, Vol. 36, No 4, 2009, pp. 7448-7457.

[15] F. A. Hansen, "Context-aware mobile hypermedia: concepts, framework, and applications", PhD Thesis, University of Aarhus, 2006.

[16] N. Weißenberg, An ontology-based approach to personalized situation-aware mobile service supply, GeoInformatica, Vol. 10, No 1, Springer, 2006., pp. 55-90.

[17] T. Strang and C. Linnhoff-Popien, "A context modeling survey", Proc. First International Workshop on Advanced Context Modelling, Reasoning and Management (UbiComp '04), 2004.

[18] E. Sanchez, T. Yamanoi, "Fuzzy ontologies for the semantic web", H.L. Larsen et. al. (eds.), FQAS 2006, LNCS, vol. 4027, pp 691-699, Springer, Heidelberg, 2006.

[19] L.A., Zadeh, "Is there a need for fuzzy logic?", Information Sciences, No. 178, 2008, pp. 2751–2779.

[20] W3C, Public Time Ontology, date, http://www.w3.org/TR/owl-time, accessed June 2009.

[21] L. Ding, H. Chen, L. Kagal, and T. Finin, Public Address Ontology, http://daml.umbc.edu/ontologies/ittalks/ address, accessed June 2009.

[22] Dey, A.K. and Abowd, G.D., "Towards a better understanding of context and context-awareness". In Proc. Workshop on the What, Who, Where, When and How of Context-Awareness, ACM Press (2000).

[23] G. Sun, J. Chen, W. Guo, and K.J.R. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs", IEEE Signal Processing Magazine, 2005, Vol.22, No.4, pp. 12-23.

[24] F. Van Harmelen, "Fuzzy logic and the semantic web", E. Sanchez (eds.), Capturing Intelligence, Elsevier, 2006.