# Knowledge-Based Systems for Ambient Social Interactions

Xiang Su[1], Ekaterina Gilman[1], Paweł Kwiatkowski[2], Tomasz Latkowski[2], Alma Pröbstl[1], Bartłomiej Wójtowicz[2], and Jukka Riekki[1]

[1] Department of Computer Science and Engineering and Infotech Oulu, University of Oulu, Finland
[2] Department of Electronics, Military University of Technology, Warsaw, Poland

**Abstract.** The development of ambient social applications brings challenges to aggregate information from heterogeneous sources, like users, physical environments, and available services. We propose a framework for aggregating information from different sources, and utilize a novel representation, Entity Notation (EN), as a starting point of connecting all information to knowledge-based systems, which offers good possibilities to support ambient social intelligence. In this paper, we present the framework, our EN representation, and an implementation of a map reminder service to demonstrate the usability of our framework.

**Keywords:** Knowledge-based Systems, Ambient Social Interactions, Entity Notation, Rule-based Reasoning.

## 1 Introduction

Nowadays, user-centric mobile applications are becoming more and more popular. They bring a new fashion of ambient social interactions among people [1]. Ambient social interactions benefit from information aggregation from the social information of users, available services, physical environment, etc. This aggregation can improve the usability and enable new functionalities for mobile social applications. However, such aggregation brings challenges due to several reasons; for example, different sources support different communication models and incompatible representations.

Semantic Web technologies enable information interchange between different sources. Heterogeneous information, like user profiles, shared community information and real time sensor measurements can be integrated into knowledge-based systems using Semantic Web technologies. Semantic Web technologies offer good possibilities to support intelligent ambient social interactions. For example, real time inference over social information, like the shared goals of a community, and physical environment information, like location, can enable ambient intelligent social interactions for individuals sharing common tasks and duties in communities. One main challenge is that social content and sensor data have no representation compatible with Semantic Web languages. Hence, it is difficult to semantically aggregate these information sources.

D. Keyson et al. (Eds.): AmI 2011, LNCS 7040, pp. 61–71, 2011.

Semantic Web technologies and social data processing require large amount of computing resources, which conflict with limited memory, computing, and communication resources available by mobile devices. Hence, most social applications have servers for storing and processing the information. However, server side communications challenge the usability of ambient social applications, as they are expected to be always available, regardless of the quality of network communications.

In this paper, we present our work towards building knowledge-based systems with Semantic Web technologies for ambient social interactions. A knowledge-based ambient social interaction framework will be suggested to bridge the gap between heterogeneous information sources. This framework offers large potential for implementing ambient social intelligence based on knowledge. Moreover, we utilize an innovated representation, Entity Notation (EN), in this framework as a starting point of connecting all information together, including sensor data, social content, Semantic Web-based inference, etc. EN is a Semantic Web languages compatible serialization, which stems from the basic semantics of RDF and ontologies. It is expressive for representing knowledge structures, and can be serialized by any sensor and application straightforwardly. EN is a promising solution to connect mobile applications with the knowledge-based systems on the server side, as it significantly reduces communication overhead. In order to verify our framework, we implement an application that reasons over social content produced by users and real-time location data to enable intelligent interactions among multiple users. We demonstrate a location-based reminder service, which presents reminders and maps to mobile users based on their needs.

The rest of this article is organized as follows. Section 2 presents related work and section 3 discusses the general framework to build knowledge-based systems for ambient social interactions. We introduce more details of EN, and our implementation details in section 4 and section 5 respectively. We conclude the paper in section 6.

## 2 Related Work

Most current mobile social applications simply extend the web interfaces of social software to mobile devices. That is, one can connect to his favorite online social network sites through his mobile phone, like Facebook Mobile [2], and LinkedIn Mobile [3]. Location-based mobile services provide interesting functionalities to facilitate social interactions. For example, Foursquare [4] is a location-based social networking software for mobile devices. It enables mobile users to explore the city, find friends, mark visited places, etc. Google Latitude [5] allows mobile phone users to share their locations with other users. Gbanga game [6] builds a virtual world according to real user locations facilitating different social interactions between users. More mobile applications, like WhozThat [7], Social Serendipity [8] and CenceMe [9] focus on finding new friends through short range communication techniques, like bluetooth.

On the other hand, applying semantics, such as ontologies, in location based services has been reported in [10] [11], but these efforts focus more on location-based navigation. Though knowledge-based systems would enable lots of intelligent functions for ambient social interactions, not much work has been done in this direction. In comparison with related work, we concentrate on building knowledge systems to aggregate information sources, which will facilitate ambient social interactions. This enhances the basic features of ambient social computing that utilize context information. Context is recognized from sensor data and by matching content and applications to current situation and needs of users. [12]. EN is suggested as a key enabler for ambient social interactions, and it is compared with other representations in [13].

## 3 A Framework for Building Knowledge-Based Systems for Ambient Social Interactions

Figure 1 presents a general framework to build knowledge-based systems for mobile social interactions. Information from heterogeneous sources, such as social information, physical environments, and available services are aggregated to knowledge-based systems. Social information includes the aspects of single person and communities. Communities are groups of people with common roles, interests or tasks. Social information can be extracted from their profiles, subscriptions, social networks, etc. Physical environments offer information about the physical surroundings of users, like location, time, noise level, temperature, etc. This information can be both quantitative and qualitative. Quantitative information is acquired by digital objects, like sensors, which measure environment and provide these measurements for knowledge bases. Qualitative information includes geographical, ethnographic and other information. Available services are other important sources for knowledge-based systems. Services provide more complex and processed data, for instance from networked sensors.

Knowledge-based systems are key enablers for ambient social intelligence. They include knowledge bases and reasoners. Knowledge bases store formal models of knowledge. In practice these knowledge models consist of a terminology part and an assertion part. The terminology part, the so called T-Box, defines formal models of concepts and states additional constraints among concepts. The assertion part, the so called A-Box, describes the concrete individuals of T-Box concepts and their relations with each other. Reasoners are software components able to infer new logical relations from the knowledge base. There are many different techniques that can be applied for reasoners, and the most popular ones are First Order Logic, Description Logic and probability theory. We utilize ontology models to represent knowledge in the knowledge base. Ontologies serve as a formal representation that machines can understand and interpret, which facilitate advanced knowledge processing, such as reasoning. Ontological knowledge bases support Description Logic (DL)-based and rule-based reasoning. Meanwhile, there are possibilities to support hybrid reasoning mechanisms when the knowledge bases of other representations are available. Hence, ontology models provide a flexible solution to support the complex relations of social interactions.

Users and applications are considered as actors in our framework. Users and applications consume deduced results produced by knowledge-based systems. However, they can also act as sources of social information, as sensors observe them in their activities.

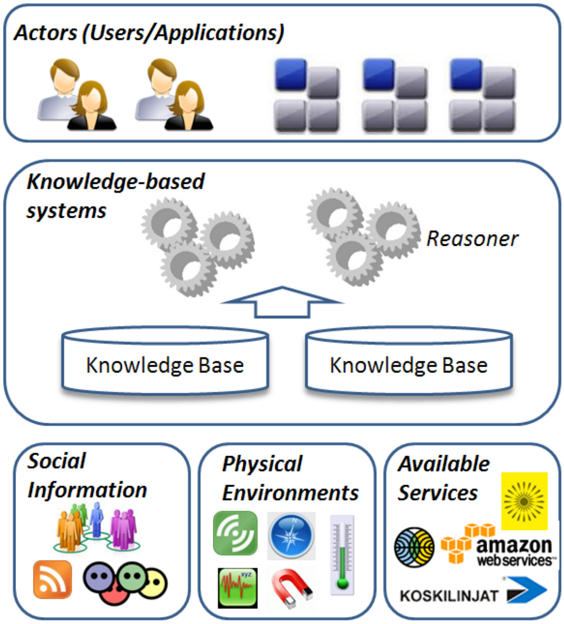


**Fig. 1.** The framework for building knowledge-based systems for ambient social interactions

Knowledge-based systems facilitate ambient social interactions by providing a formal grounding for all information sources. When such a common formal grounding is available, user specific social information, sensor produced data, and all other information can be aggregated based on the same grounding. That is, knowledge-based systems establish the common meaning of entities and their interactions. Interacting entities are able to share not only data, but semantics, for ambient social intelligence. Knowledge-based systems provide mechanisms to identify implicit logical connections and enable semantic interoperability straightforwardly and unambiguously.

Knowledge-based systems require a lot of computing resources from the hosting devices. A knowledge base can consume large amount of memory, which is not available from mobile devices, while reasoning is resource consuming process and only a limited set of currently available mobile devices can perform this operation constantly. Hence, it is a common practice to build big knowledge bases and complex reasoning processes on the server side. However, there is a possibility to deploy partial knowledge base and affordable reasoning on mobile devices. This will enable faster response time, enhanced privacy, and avoidance of the uncertainty, caused by communication issues of environments.

## 4 Entity Notation

Aggregation of heterogeneous information from different sources requires a powerful and flexible representation. In this section, we suggest utilizing EN [13] [14], to interconnect all information. EN was designed as a lightweight serialization of knowledge models. It is expressive to serialize ontologies and RDF models straightforwardly, and on the other hand, it is so simple that any sensor and application can compose EN packets. EN is compatible with Semantic Web languages, like RDF/XML and OWL. Hence, it can be interoperated with popular social contents, like RSS and FOAF ontology easily. Sensors can compose EN packets with minimal computing resources, and still the data can be transferred into RDF knowledge. Moreover, EN can be the payload of underlying protocols, like HTTP; hence, information from available services can be expressed and transferred.

Generally, an EN packet depicts an entity and its relationships with values and other entities. To fulfill the expressibility and lightweight requirements, we define two EN formats: the complete format, and the short format. The complete format has enough expressibility and has the following format:

```
EntityType EntityID
 PropertyName PropertyValue
 ...
 PropertyName PropertyValue
```

We utilize square brackets and angle brackets to identify the level of knowledge an EN packet should be mapped to. When an entity packet is wrapped in square brackets ([ and ]), this EN packet should be transferred to A-Box knowledge. When an entity description is wrapped in angle brackets (< and >), this EN packet should be transferred to T-Box knowledge. Here are two examples of EN complete packets. In the first packet, Alice is sharing a task with Bob, which is to pick someone up at University of Oulu (social information). The second packet represents locationsensor measurement of Alice (physical environments information).

```
[http://ee.oulu.fi/o#SharedTask http://ee.oulu.fi/o#pickupSharedTask101
http://ee.oulu.fi/o#ownerID  "Alice"
http://ee.oulu.fi/o#peerID  "Bob"
http://ee.oulu.fi/o#interestingPlace
http://ee.oulu.fi/o#universityofoulu]


[http://ee.oulu.fi/o#LocationSensor http://ee.oulu.fi/o#locaSensor767
http://ee.oulu.fi/o#ownerID  "Alice"
http://ee.oulu.fi/o#longitude  "25.468"
http://ee.oulu.fi/o#latitude "65.058"]
```

Lightweight short packets can support resource-constrained devices and slow communication links. The short EN format uses templates and prefixes to shorten packets. The basic idea is that a template contains a description of the constant part of a sequence of EN packets and placeholders for variable items. Prefixes are used to shorten URI references. Similar to complete packets, square and angle brackets are utilized to wrap the following descriptions depending on the level of knowledge, and we utilize UUIDs to identify templates:

```
UUID PropertyValue  ... PropertyValue
```

With templates and prefixes, we can shorten the examples above to:

```
[urn:uuid 76eac2 "Alice" "Bob" EE#universityofoulu]

[urn:uuid 539ea2 "Alice" "25.468" "65.058"]
```

Compared with other knowledge serializations, EN packets can be very compact, hence the communication overhead over server and mobile client communication links can be significantly reduced. This enables building knowledge-based systems on the server side, and still keep low overhead as possible.

## 5 Event Map Implementation for Ambient Social Interactions

To prove the usefulness of our designed framework we have implemented an ambient social application, which provides the reminder support for mobile users. This application enables rendering mobile maps and reminders to users according to ambient information, including fixed points of users' interests (PoIs), the geographical locations of mobile devices, and shared tasks of users. Application is executed in the background and the user can perform normal mobile phone activities. Ambient maps with reminders come to the foreground when there is important information to be shown to the users. We demonstrate two scenarios in our implementation. In the first one two persons are sharing a task of buying pizza for them at noon, and the one who is closer to the pizza restaurant (that is the PoI) will get an alert with a corresponding map to visit the restaurant. The second scenario tells about family. The wife specifies a shared task with her husband to pick up their kids from a school (that is the PoI). The couple has decided to pick up their kids as early as possible, so at the predefined time, the one who is closer to the school gets a reminding alert with a map on his or her mobile phone to pick up their kids.

Figure 2 presents the architecture of the implemented system, which consists of two main parts: a client side and a server side. The client side is a Java ME application, consisting of a task generator and a GPS receiver to process location data. With task generator users can specify their preferences or settings about which reminder to show, when to show, and whom to share the task with. Both location data and specified tasks are sent with short EN packets to the server to minimize communication payloads. An example of these packets are presented in section 4. The server side consists of an EN decomposer, a domain ontology, a

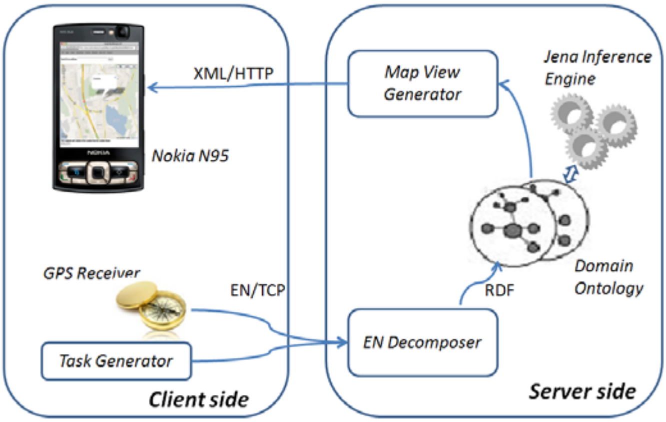


**Fig. 2.** Implemented architecture of ambient social interactions

reasoning engine and a map view generator. The EN decomposer transforms EN packets into RDF statements. These RDF triplets are added into the domain ontology and trigger the inference engine. We define a domain ontology, which includes predefined PoIs (such as restaurants and schools), user profiles, shared tasks, and other facts relevant to the system. The inference engine is a rule-based system which constantly checks if any task defined by any user can be executed and triggers corresponding rules. Rules are considered as instructions for actions that can be applied to a certain situation. If the task is executed, the map view generator renders the map according to the task settings. This map is sent to the specific person at specified time.

Our implemented prototype fully realizes the framework presented in Figure 1. We utilize user locations and shared tasks (social information in communities) as suppliers to the knowledge-based system. As can be seen from Figure 2, the knowledge-based system in our implementation consists of domain ontology and inference engine components. The domain ontology is a small knowledge base containing T-Box and A-Box knowledge. Our reasoning engine is a Jena rule-based system and it utilizes pre-designed rules to infer conclusions from the premises. Users act in the environment and operate with the application.

Table 1 shows some rules defined for our application scenarios. The left column describes the rule in natural language and the right column presents the Jena rules, applied to the domain ontology.

The left picture in Figure 3 is a screenshot from an emulator, which demonstrates the picking up kids scenario. A red mark specifies the target location and a blue mark specifies the current location of the user. Also a textual reminder of this task is shown to the user. The right picture captures the real user interaction with the application during our field test, which is a shared task for visiting a pizza restaurant.

**Table 1.** Social Interaction Rules for Event Map Implementation

| Rule description | Reasoning rule |
| --- | --- |
| Alice creates the task and shares it with Bob. This task is supposed to be executed for one (Alice or Bob) who is closer to certain point of interest. This rule checks for the shared task (?sharedTask) whether Bob (?user) is closer to the point of interest (?place). | [rule1_check_closest_user:<br>(?sharedTask rdf:type sw:SharedTasks)<br>(?sharedTask sw:hasPersonToGo ?person)<br>(?sharedTask sw:smallestDistance ?smallestDistance)<br>(?sharedTask sw:hasSharingMember ?user)<br>(?user sw:X ?userX)(?user sw:Y ?userY)<br>(?sharedTask sw:hasPlace ?place)<br>(?place sw:X ?placeX)(?place sw:Y ?placeY)<br>countDistance(?userX, ?userY, ?placeX,<br>?placeY, ?distance)<br>lessThan(?distance, ?smallestDistance)<br>noValue(?smallestDistance sw:firedFor2 ?sharedTask)<br>—><br>remove(1,2)<br>(?sharedTask sw:smallestDistance ?distance)<br>(?sharedTask sw:hasPersonToGo ?user)<br>(?smallestDistance sw:firedFor2 ?sharedTask)<br>hide(sw:firedFor2)] |
| This rule checks if the current time (?currentTime) is within the shared task validity time (?validity), if it is then the task is ready to be executed, hence its transmission property becomes "transmission_succeded". Also, the new time for task execution is set according to task repetition property. | [rule2_set_transmission_for_shared_task:<br>(?sharedTask rdf:type sw:SharedTasks)<br>(?sharedTask sw:time ?time)<br>(?sharedTask sw:transmission 'no_transmission')<br>now(?currentTime)<br>validityTime(?time, ?validity)<br>le(?time, ?currentTime)<br>ge(?validity, ?currentTime)<br>(?sharedTask sw:repetition ?repetition)<br>setTime(?repetition, ?time, ?newtime)<br>(?sharedTask sw:hasPlace ?place)<br>(?place sw:X ?placeX)(?place sw:Y ?placeY)<br>(?sharedTask sw:hasPersonToGo ?user)<br>(?user sw:X ?userX)(?user sw:Y ?userY)<br>noValue(?user sw:firedFor3 ?task)<br>—><br>remove(1,2)<br>(?sharedTask sw:time ?newtime)<br>(?sharedTask sw:transmission 'transmission_succeded')<br>(?user sw:firedFor3 ?sharedTask)<br>hide(sw:firedFor3)] |
| This rule gathers all necessary information for the shared task (?sharedTask) which is ready to be executed (transmission property equals "transmission_succeded"). This information is obtained to generate the map with the reminder. | [rule3_send_map_for_shared_task:<br>(?sharedTask rdf:type sw:SharedTasks)<br>(?sharedTask sw:time ?time)<br>(?sharedTask sw:transmission 'transmission_succeded')<br>(?sharedTask sw:repetition ?repetition)<br>(?sharedTask sw:hasPlace ?place)<br>(?place sw:X ?placeX)(?place sw:Y ?placeY)<br>(?sharedTask sw:hasPersonToGo ?user)<br>(?user sw:X ?userX)(?user sw:Y ?userY)<br>(?sharedTask sw:comment ?comment)<br>—><br>mapInfo(?sharedTask,?comment,?user,?userX,<br>?userY,?place,?placeX,?placeY,?repetition,?time)] |

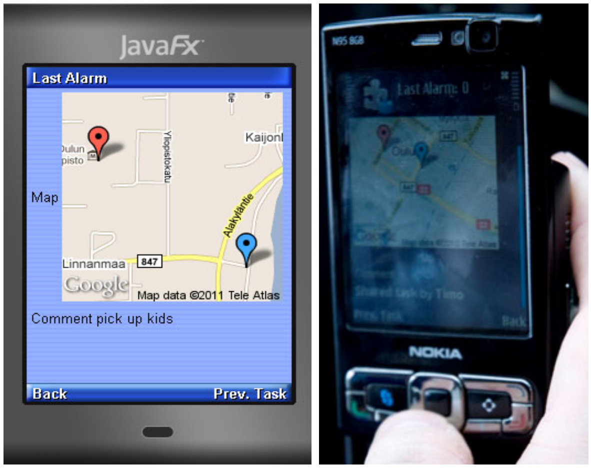


**Fig. 3.** Application screenshots

We implemented our client on Nokia N95 mobile phones and tested it with panOulu open access network. The user test has demonstrated the usability of our system. The application runs in the background, allowing users to perform their daily routine and appears at the foreground only if the execution time of the task (which was specified by the user) is reached. The concepts of tasks, points of interests and the logic of application were clear to the users and they could easily manage tasks by themselves. The field test also demonstrated that users are eager to share their location and task information *"as long as I [Test user] am in control what and when and to whom I share it with"*. The overall application was considered as a nice system which actually could be used as *"part of a calendar application"*.

## 6 Discussion

In this paper, we presented our work towards building knowledge-based systems for ambient social interactions. We discussed a general framework to enable Semantic Web technologies in ambient social interactions and details of EN. Scenario implementations demonstrate the usability of our framework. The aggregation of different information sources for ambient social intelligence is challenging since it requires the confluence of previously separated lines of research. We expect that Semantic Web technologies are promising solutions to connect heterogeneous information sources. In this article, we proposed a general framework and EN that serve as a starting point for advanced ambient intelligence features.

By their nature, ambient social applications must be available for users everytime and everywhere. Moreover, they are expected to be lightweight (consume few resources), allow normal mobile phone routine (like, phone calls), and respond fast. Knowledge-based systems facilitate different social behavioral interactions, even though it is a challenge to utilize the server back end communication for mobile applications, because of delays, connectivity, etc. We consider EN as a good solution to minimize communication overheads. Moreover, EN is an ideal candidate for playing important roles as an underlying representation for different information sources. It is expressive for representing information from social and physical contexts.

Building knowledge-based systems for social applications brings clear separation of the functionality between system components. That is, main advantage of this approach is that we isolate the application logic from the system. The application logic is mainly performed by rules which can be easily modified and extended on the fly, without a recompilation of the system. Hence, the functionality of the system can be easily changed. This is a very important issue especially for mobile applications, where users are not so eager to install application updates.

Our current implementation is limited in its consideration of privacy and security issues, because we reason all private information on the server side. Moreover, we could consider enabling more dynamic interactions, for instance allow users to trigger tasks when a certain friend is nearby, which is not supported in the current implementation.

As future work, we will consider deploying knowledge-based systems on mobile devices. This is one big step for realizing privacy-protected and fast ambient social interactions. Moreover, we will consider other possibilities of social information; for example, aggregating FOAF ontology and RSS feeds to enable intelligent functions for ambient social intelligence.

**Acknowledgments.** This work was funded by MOTIVE research project program of the Academy of Finland. The first author would thank the funding from NOKIA Foundation, and the second author would thank the funding from GETA (Graduate School in Electronics, Telecommunications and Automation). We would also thank Timo Saloranta and Dazhuo Li for being our volunteers of user tests.

## References

1. de Ruyter, B.: Socail Interactions in Ambient Intelligent Environments, PhD thesis, Eindhoven University of Technology (2010)
2. Facebook Developer Documentation, `http://developers.facebook.com/docs/`
3. LinkedIn Professional Network, `http://linkedin.com`
4. Foursquare, `https://foursquare.com/`
5. Google Latitude, `https://www.google.com/latitude`
6. Gbanga, `http://gbanga.com/`
7. Beach A., Gartrell M., Akkala S., Elston J., Kelley J., Nishimoto K., Ray B., Razgulin S., Sundaresan K., Surendar B., Terada M., Han R.: WhozThat? Evolving an Ecosystem for Context-Aware Mobile Social Networks. IEEE Network, 50–55 (July/August 2008)

8. Eagle N., Pentland A.: Social Serendipity: Mobilizing Social Software. IEEE Pervasive Computing, 28–34 (April/June 2005)
9. Miluzzo, E., Lane, N.D., Eisenman, S.B., Campbell, A.T.: CenceMe – Injecting Sensing Presence into Social Networking Applications. In: Kortuem, G., Finney, J., Lea, R., Sundramoorthy, V. (eds.) EuroSSC 2007. LNCS, vol. 4793, pp. 1–28. Springer, Heidelberg (2007)
10. Tryfona, N., Pfoser, D.: Data Semantics in Location-Based Services. In: Spaccapietra, S., Zimányi, E. (eds.) Journal on Data Semantics III. LNCS, vol. 3534, pp. 168–195. Springer, Heidelberg (2005)
11. Kolomvatsos K., Papataxiarhis V., Tsetsos V.: Semantics Location Based Services for Smart Spaces. In: Metadata and Semantics, Part 7, pp. 515–524. Springer Press, Heidelberg (2009)
12. Vos, D.H., Haaker, T., Teerling, M.: Consumer Value of Context Aware and Location Based Mobile Services. In: 21st Bled eConference eCollaboration: Overcoming Boundaries through Multi-Channel Interaction, pp. 50–62 (2008)
13. Su, X., Riekki, J., Haverinen, J.: Entity Notation: Enabling Knowledge Representations for Resource-Constrained Sensors. Personal and Ubiquitous Computing (accepted, 2011)
14. Su, X., Riekki, J.: Transferring Ontologies between Mobile Devices and Knowledge-based Systems. In: 8th IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, pp. 127–135. IEEE Press (2010)