A Unified Scalable Model of User Localisation with Uncertainty Awareness for Large-Scale Pervasive Environments

Petr Aksenov, Kris Luyten, Karin Coninx
Hasselt University – transnationale Universiteit Limburg
Expertise centre for Digital Media – IBBT
Wetenschapspark 2, BE-3590 Diepenbeek, Belgium
{petr.aksenov, kris.luyten, karin.coninx}@uhasselt.be

***Abstract*— Localisation has become a standard feature in many mobile applications. Numerous techniques for both indoor and outdoor location tracking are available today, providing a diversity of ways positioning information can be delivered to a mobile application (e.g., a location-based service). Such factors as the variation of precision over time and covered areas or the difference in quality and reliability make the adoption of several techniques for one application cumbersome. This work presents an approach that models the capabilities of localisation systems and then uses this model to build a unified view on localisation, with special attention paid to uncertainty coming from different localisation conditions and its presentation to the user. We discuss technical considerations, challenges and issues of the approach and report about a user study on users' acceptance of the suggested behaviour of an application based on the approach. The results of the study showed the feasibility of the approach and revealed users' preference towards automatic but yet informed changes they experienced while using the application.**

Keywords-adaptation techniques; localisation systems; scalability; uncertainty; user-awareness

I. Introduction

Location is often considered the most important type of context [28] and, as discussed by Cheng et al. [14], exploiting the knowledge about location is much more beneficial when it is ubiquitously available. The rapid development of technology leads to better and improved techniques to determine location so that the number of places where a user can be located also increases. Indeed, mobile devices are used in many places today: in the office, at a parking lot, in a supermarket, in the city centre, at home, etc. As a consequence, location-aware applications and location-based services have become very popular recently and their market still continues to grow (e.g., Foursquare [1], Gowalla [3]). However, the resulting flexibility and ubiquity bring in their own issues. According to Banerjee et al. [10], two major types of location tracking techniques can be distinguished: infrastructure-based (e.g., Ubisense [7], Place Lab [19]) and peer-based (e.g., Virtual Compass [10]). Each of them can provide quite a different set of location data but even such simple parameters as how often the measurements are provided or how precise they are can influence the application's behaviour. Therefore this diversity can create limitations for applications that want to make use of different tracking technologies. For example, existing frameworks may not support all of the available localisation systems or have little to no support for subtle differences in location data provided by different tracking systems. Changes in location tracking systems should be transparently handled according to the properties of a specific tracking system.

In order to integrate these changes in location into the application's behaviour, we need a unified way of accessing the location data provided by arbitrary tracking systems, as well as the metadata related with the tracking system. Since location determination is error-prone and generally incomplete, it often comes with uncertainty about the exact location and even about the reliability of the tracking system itself.

We present a strategy for handling changes in the (re)presentation of location according to the available localisation over time so that the application reflects them and behaves appropriately in each such situation. We address how the diversity in tracking systems (and thus location data) can be handled at any place and at any moment of time. We present an approach that models the properties of localisation systems and uses this model to build a unified view on localisation throughout a large-scale pervasive environment. Our approach takes into account any location tracking system in the environment and only requires the availability of the description of these systems at design-time. We explicitly consider *uncertainty of location* in our model and establish further use of the uncertainty information in applications. In short, our approach should increase awareness on the uncertainty of the position information presented to the user.

In the sections to follow, we discuss technical considerations, issues and challenges of the approach. We also report about a user study performed on a prototype location-aware application based on the proposed approach. The results of the study show that our approach performs well and the users react positively on learning about the changes in their localisation. This supports our assumption that users prefer to stay aware of the nature of the uncertainty coming from localisation and provided us with ample grounds to continue our work.

II. Related Work

An increasing role of pervasive and ubiquitous computing has given rise to research about problems in which multiple location tracking technologies are involved. In our earlier work on variability of location sensing [9], we discussed how spatial information can be interpreted and reorganised under different location tracking conditions. However, the reported sample use-case was performed in a simulated environment, which may not always be the best choice [15].

On the general level, LOC8 [26] is a programming framework that models location and localisation systems as an ontology, providing a querying mechanism to support the use of location by pervasive applications. A set of presented use cases illustrate how the framework is used, accompanied by a discussion of the efforts that the creators of location-aware applications should take into account with respect to supporting different positioning systems. The fundamental approach presented by Coronato et al. [16] is similar: a spatial model is described as an ontology which is used to reason about a better location among the available ones as well as to resolve possible location conflicts. However, no further details of the localisation system are considered in that work. The authors of Streamspin [18], a platform for the creation and delivery of location-based services, addressed key aspects of seamless handover between GPS for outdoors and wifi-based localisation for indoors based on the fact whether either system was detectable. The results of their experiments clearly indicated the need for such kind of support. MiddleWhere [24] is an example of approaching the support of location information as an independent service separated from the rest of the application. The suggested infrastructure is based on a model of spatial relations and reasoning about location is supported in that separate service. However, the main focus was on fusing location data whenever possible in order to provide more accurate positions.

Addressing location uncertainty has also been an important research problem. In particular, researchers have been trying to understand how the users of location-aware applications can and should be informed about it. Thus, Dearman et al. [17] found out that knowing about location error aided users in their navigational tasks with a map-based application. Benford et al. [12] presented an extended discussion on the ways to approach positional uncertainty. The authors suggested that uncertainty of user location should be addressed and categorised and they proposed five different strategies of dealing with it, depending on the context of application use. However, it was concluded in the follow-up research that a good representation of uncertainty is still a challenge [22]; and Patel et al. [23] stated that it may even be more challenging when developing general-purpose applications for non-experts.

The two trends – supporting the diversity of localisation technologies and providing user-awareness about the uncertainty these technologies have – seem to be of considerable interest to the research community and form important directions in developing future location-aware applications for pervasive environments. However, their combination seems to be scarcely presented, and by placing them under the same roof, we tie the support of localisation systems with their limitations revealed to the user.

III. The Approach

Our approach comprises two principal components. The Unifying Component (see Section III.A) takes care of receiving location measurements and converting them into the common format of location data thus making them ready for further processing. The Location Processing Component (LPC, see Section III.B) performs the actual processing of received updates based on the involved reasoning methodology. There is also a visualisation component included in our system, which presents location updates to the user and visualises location metadata (such as the reliability, history, etc). We will show how including metadata in the visualisation leads to more awareness on the status of the location tracking.

All the communication in the environment uses W2P, a message-oriented and HTTP-based peer-to-peer communication system [27]. Since internet-capable mobile devices have become quite popular and staying connected to the web is possible almost everywhere in urban areas these days, W2P turns out to be a convenient and good choice for easy and stable delivery of location updates. Furthermore, HTTP is a ubiquitous lightweight protocol accessible from

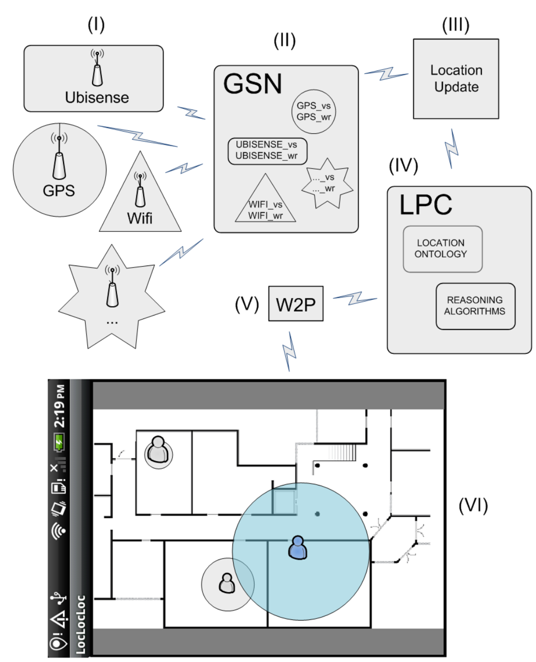

Figure 1. GSN (II) handles all incoming location data (I) and produces a location update (III). LPC (IV) processes the update and informs the users who get the new update visualised on their devices (VI). All communication in the environment happens via W2P (V).

almost all mobile devices nowadays. Using HTTP in a peer-to-peer topology also allows us to decentralise location tracking services which is consistent with our aim to connect with both local location tracking providers (e.g. Ubisense) and centralised services (e.g. Google hotspot maps).

The structure of the system is outlined in Fig. 1 and is discussed in details in the remaining part of this section.

A. The Unifying Component

In general, a different localisation technology has a different set of parameters (illustrated by blocks of different shapes in Fig. 1.(I)), different quality of location [13], etc. For example, a location tracking system can produce 2D coordinates of an object every 10 seconds or it can only provide an update if the object is within the "visibility" distance from a certain sensor so that the time of the next update cannot be known at all (e.g., a passive RFID tag used for localisation does not have a predefined update rate). Furthermore, the location data that are delivered can be of an arbitrary structure. Nevertheless, all data have to be correctly interpreted by the framework.

We have created a software layer that converts different location formats into a common unified format. For this, we assume that each localisation system that comes into play can provide a semantic description for the data it generates, which is used by our unification process. We use the Global Sensor Networks (GSN) framework [8] for this purpose (Fig. 1.(II)). GSN is a middleware for processing diverse sensor data. It supports introducing new location providers at run time and has proved to work successfully in multiple occasions. It requires that each location provider has a so called "virtual sensor" containing an XML description of the parameters. This parameterisation is then used by a Java-wrapper that handles the data the provider sends. For our approach to work, the processing elements we introduced need to be made available during the setup phase of a localisation system since these are key elements in transforming the data into the unified format.

We implemented a template wrapper that has the W2P communication established. With this template, creating the required wrapper for a location provider would only require specifying the original location data this provider delivers, in the form understood by W2P. An example of such modification is shown in Fig. 2. The "virtual sensor" can be customised similarly, and an example is shown in Fig. 3. Note that if, for example, a localisation system does not have an update rate, this field may simply be skipped and left empty. Creating the virtual sensor and the wrapper is a one-time activity and, once completed, causes no hurdles at run-time.

Additionally, GSN can have multiple instances running in the environment and communicating as peers. Therefore location measurements from a particular location provider can be correctly processed by the corresponding instance of GSN this provider is attached to, thus providing the required scalability.

```
postStreamElement(
        w2p.getLocationEvent().getTimestamp(),
        w2p.getLocationEvent().getTagId(),
        w2p.getLocationEvent().getLocSysId(),
        w2p.getLocationEvent().getUpdateRate(),
        w2p.getLocationEvent().getX(),
        w2p.getLocationEvent().getY(),
        w2p.getLocationEvent().getError(),
        ...
        ...
);
```

Figure 2. An extract, in pseudo-code, from a sample Java-wrapper required by GSN

```
...
...
<processing-class>
        <class-name>gsn.vsensor.W2PVirtualSensorProcessor</class-name>
        <init-params />
        <output-structure>
                <field name="timestamp" type="varChar(13)"/>
                <field name="tag_id" type="varChar(15)"/>
                <field name="locsys_id" type="varChar(8)"/>
                <field name="update_rate" type="Integer"/>
                <field name="xcoord" type="double"/>
                <field name="ycoord" type="double"/>
                <field name="error" type="double"/>
        </output-structure>
</processing-class>
...
...
```

Figure 3. An extract from a sample virtual sensor required by GSN

B. The Location Processing Component

A unified location update created by GSN (Fig. 1.(III)) is then passed to the location processing component (LPC, see Fig. 1.(IV)). LPC is a custom application that processes incoming location updates in order to decide about the location and the status of the tracked objects. It has an architecture that can be extended to include any existing algorithms for location processing (e.g., location fusion, quality of location [13]).

We use an ontology to model the semantics of location data and tracking systems. Ontologies proved to be useful for describing location information and reasoning upon it [21]. Our ontology adheres to the Web Ontology Language (OWL) [6] and consists of two parts: the semantics of a localisation system itself and the semantics of a location update it generates (an example of our ontology is shown in Fig. 4). The ontology is used to describe and address the details of various location providers and their location updates and the XML description defined in GSN is used as a protocol for registering a new system or for notifying about the updates. The syntax of OWL is based on XML therefore matching it with the GSN protocol is easy.

We involved a simple set of parameters from each localisation system so far, such as the update rate and the precision of its location updates, and use the error and age of a location update to decide on which location update to use. The age depends on the update rate so that the most recent location is considered obsolete if it passes two scheduled updates or the universal threshold of 10 seconds, whichever happens first. This sort of comparison is simple but the simplicity was justified by the informative nature of the initial evaluation which aimed at guiding the further development of the system and identifying the parts that need to be changed. In general, the reasoning on location involves many more parameters than we discuss here, including those from the user model (walking speed), the

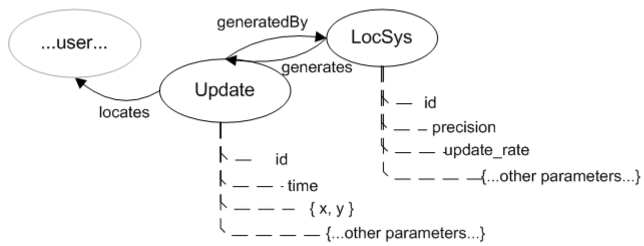

Figure 4. An example of an ontology to describe localisation systems. It includes a separate concept of a single location update, i.e. the location data the system provides.

task model (attending a meeting), the topology of the environment (locations on water are invalid), etc. Therefore we designed our reasoning mechanism to be extendable, and an external ontology handling component can be readily added to the LPC's functionality. For example, the LOC8 framework [26] could be integrated when it is released.

C. The Visualisation Component

After a location update has been processed by LPC, the client applications and thus their users have to be notified about the result. Possible outcomes from LPC include, amongst others, a change in position, an increased or decreased reliability of the position information or a different error in the position information.

The modelling and reasoning part of our approach is accompanied by a visualisation component that informs the user about the changes (see Fig. 1.(VI)). Inspired by Benford et al.'s four 'states of being' of a mobile user [12], we introduced four different states of awareness about the status of location tracking: (a) a location update is received with an initial positioning error, and the dot in the centre starts pulsating (Fig. 5.(a)); (b) the precision area extends as a function of the time elapsed since the last update.

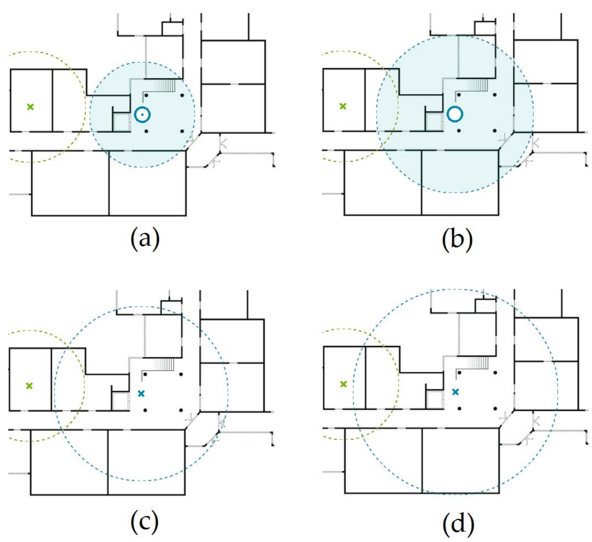

Figure 5. Visualisation of the status of a user's location tracking considers four different states of awareness: (a) a regular update is received; (b) an update is missing; (c) location tracking is possibly unavailable; (d) location information is outdated (visualisation is static).

Moreover, if the threshold for receiving the next update has been exceeded, the dot disappears (Fig. 5.(b)); (c) if the second threshold of the update rate has been exceeded, the marker transforms into a cross and the precision area is no longer shown replaced with a single dashed-line (Fig. 5.(c)); (d) the previous state extends further until the maximum precision threshold (currently set to 20 metres) is reached Fig. 5.(d).

We use a circle-based representation of user location on a map. Such visualisation incorporates all data and metadata we currently want to show about the position, and it can be often met in map-based applications (e.g., [2],[17]). In general, other shapes that depend on the nature of the system and take into account the topology of the environment may suit better [11] but determining the most appropriate visualisation is beyond the scope of the current work.

Note that the visualisation proposed in Fig. 5 equally applies to moving and stationary users and reflects their location tracking state in both cases.

IV. EVALUATION

A. User Study

Our user study pursued two goals. The first goal was to evaluate the feasibility of the proposed approach in general and to explore how changes and transition between available localisations are functioning. The second goal intended to test the other part of the approach and to get users' opinion on whether the awareness about changes influences the understanding and satisfaction of using the application.

We involved three location detection systems in our study: Ubisense, GPS, and a wifi-based weighted centroid algorithm similar to the ones used in [14] and [20]. Each system covered a different area and provided a different level of precision. Possible overlaps were minimal and existed only in the Ubisense coverage area (zone B in Fig. 6) where the wifi-based localisation produced a location but was obviously less precise or reliable.

The developed application had two modes of awareness about the detected position: one mode showed only the determined position without any additional information whereas the other mode also displayed the positioning error.

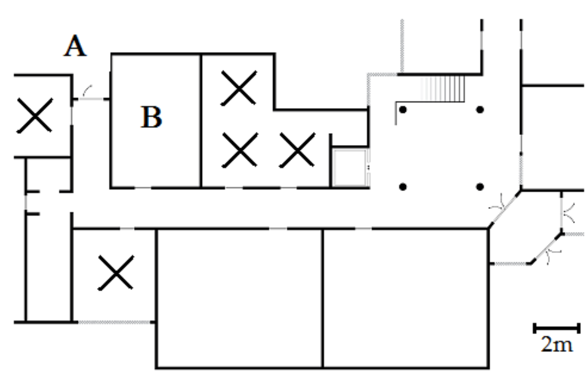

Figure 6. A part of the map of the involved area. Locations around zone A are tracked by GPS, around zone B by Ubisense, and the rest of the area is controlled by a wifi-based tracking. The areas marked with a cross cannot be walked into.

Besides, we also introduced two modes of reflecting the change of the active tracking system. In one mode, the user was notified about a change (a pop-up message was displayed on the screen) whereas no feedback was provided in the second mode. Each mode of awareness about positional uncertainty was then coupled with each mode of notifications about changes in the active tracking. This resulted in four different modes of presenting the information: (A) both uncertainty and notification; (B) only uncertainty; (C) only notification; (D) none of them.

We invited 8 participants, 5 male and 3 female, between 20 and 34 years old. Seven had an IT background and one was a graphic designer, all being employees, visitors or computer science students of our university. We asked the participants to rank their experience of 1) using map-based applications for navigation, and 2) working with smart-phones, on a five-point Likert-type scale: 0-none, 1-a bit, 2-some, 3-quite a bit, 4-a lot of. The average values were 2.1 and 2.6, respectively.

The task we asked our participants to do in each mode was the same: carrying an HTC Desire Android smart-phone with our application running in one of the four modes, the participants were asked to walk throughout the area shown in Fig. 6. As they walked, the participants had to observe the presented information that complied with the currently active presentation mode. There were between 1 and 3 objects tracked at every certain moment and. The participant was one of the objects; the other two were the experimenters. Upon completing the walking part, the participants were asked to evaluate their experience of using the application in each mode by the following relevant criteria of the NASA-TLX questionnaire [4]: *mental demand, effort, performance,* and *frustration*. The participants were also invited to provide additional comments to support the given rankings. In the end of the experiment, we also asked the participants to rank all four modes together, from best to worst. The test took around 40 minutes.

B. Results

In analysing the provided scores, we mainly focused on the pairwise comparison of the two presentation modes, with and without notifications, for each type of the visualisation of the position, i.e. A-B and C-D. Overall, 64 comparisons were made, of which participants 6 and 8 evaluated 16 pairs reporting to feel no difference between the two modes within one pair. The other participants evaluated all together 48 pairs, 19 of which were ranked as giving no change, 27 times the informed case was preferred, and 2 times the case with notifications disabled was ranked higher. Fig. 7 compares these numbers in a chart. It is also worth noting that for the situations where the informed mode was preferred, the average difference was (5.12±2.38) gradations of the 21 gradations of the TLX-scale whereas the two non-informed modes were ranked only 1-2 gradations higher than their informed compartments within the pair.

TABLE I shows the ranking the participants gave to the four use cases, from 4 (best) to 1 (worst). Six of the eight participants ranked the application mode with both uncertainty and notification active during navigation as the preferred one. It is interesting to note that P8, who preferred not to receive notifications, was the only one who gave the maximal grade of 4 to his experience with both map-based navigation and smart-phones.

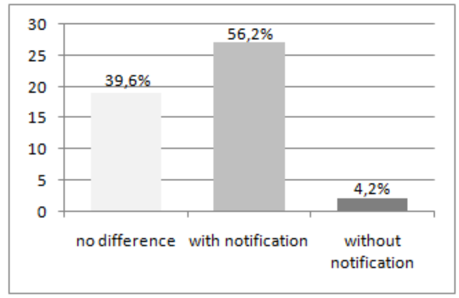

Figure 7. An informed switch between location tracking systems (i.e. with explicit notification) was the preferred option within the same visualisation.

TABLE I. APPLICATION MODES (A), (B), (C), (D)* RANKED BETWEEN 4 (BEST) AND 1 (WORST) BY EACH PARTICIPANT

	(A)	(B)	(C)	(D)
P1	4	2	3	1
P2	4	3	2	1
P3	4	2	3	1
P4	2	1	4	3
P5	4	3	1	2
P6	4	3	2	1
P7	4	3	2	1
P8	3	4	1	2
avg.	3.63	2.63	2.25	1.5

V. DISCUSSION

The main purpose of our evaluation was to serve as a test for the hypotheses we started from and to verify the feasibility and evaluate the ease of use of the proposed approach. Besides, we also wanted to identify potential flaws and issues prior to developing further extensions and performing an extended field study when the cost of a delay or an overlooked fact is naturally much higher and retaking is quite problematic and also expensive [15]. Therefore we performed our initial evaluation on a small scale using a prototype with a set of basic properties, and the feedback we received was generally inspiring.

Previous research has investigated and confirmed people's preference to receive information about positional uncertainty. In the research described in this paper we additionally allowed the users to know the reasons for this uncertainty. We did not intend to test the degree of effect on user awareness. Instead, we wanted to learn whether there would be any effect on it, and the obtained results showed a clear trend in the users' choice to stay aware about the cause of the changes they experienced.

Our evaluation focused on map-based navigation throughout and in the vicinity of a building. However, in general, the amount and type of location information users are willing to share or would like to receive varies considerably. The suitability of collecting and providing such information depends, among other factors, on the tasks the users are performing and it can be affected by even a small change in the task [25].

VI. Conclusions and Future Work

We have presented an approach to cope with the diversity of available systems for user localisation at run-time. Our framework integrates different location tracking systems based on the semantic description (ontology) they provide. The presented approach of location gathering in combination with the peer-to-peer network topology we use ensures scalability of the approach to large-scale environments (e.g., a typical city). An important aspect of the approach is the conversion of an arbitrary location update into a unified format which is then used by a separate *Location Processing Component* (LPC) that deals with uncertainty inevitably present in localisation. We explicitly reveal this uncertainty of the provided positions and changes in the degree of uncertainty to the user. The results of our initial evaluation showed users' general acceptance of informed automatic changes in their localisation.

The future work focuses on two aspects. First, we are currently developing an extended user study with more participants and in a larger area. The study will investigate different options of presenting location tracking conditions and analyse their influence on user performance in navigation-based tasks. Second, we want to extend the ways location providers can be integrated. Possible solutions include developing mapping components to the currently required Java-based wrapper or accomplishing mappings for the different location tracking systems on the ontology level. The ontology mappings allow discovering the similarities and differences between systems automatically with an ontology reasoner such as Pellet [5].

Acknowledgment

Part of the research at EDM is funded by EFRO (European Fund for Regional Development) and the Flemish Government. Funding for this research was also provided by the Research Foundation – Flanders (F.W.O. Vlaanderen, project CoLaSUE, number G.0439.08N.).

References

[1] Foursquare, http://foursquare.com

[2] Google Maps for mobile, http://www.google.com/mobile/maps

[3] Gowalla, http://gowalla.com

[4] NASA TLX, http://humansystems.arc.nasa.gov/groups/TLX/

[5] Pellet: OWL 2 Reasoner for Java, http://clarkparsia.com/pellet

[6] Web Ontology Language (OWL), http://www.w3.org/2004/OWL/

[7] Ubisense, http://www.ubisense.net

[8] K. Aberer, M. Hauswirth, A. Salehi. Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks. In *Proceedings of the 2007 Int Conf on Mobile Data Management* (MDM '07). IEEE Computer Society, Washington, USA, 198-205

[9] P. Aksenov, K. Luyten, K. Coninx: Coping with variability of location sensing in large-scale ubicomp environments. *In Proc. SEACUBE 2009,* pp. 1–5

[10] N. Banerjee et al. Virtual Compass: relative positioning to sense mobile social interactions. *In Proc. 8th International Conference on Pervasive Computing*, LNCS 6030, pages 1--21, May 2010

[11] J. Baus, C. Kray. Frames of Reference, Positional Information and Navigational Assistance. In *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference*, 2002, AAAI Press 461-465

[12] S. Benford et al. Can you see me now?. *ACM Trans. Comput.-Hum. Interact.* 13, 1 (March 2006), 100-133

[13] M. Berchtold, T. Riedel, C. Decker, M. Beigl, C. Bittel. Quality of Location: Estimation, System Integration and Application, *The 5th International Conference on Networked Sensing Systems* (INSS 2008) June 17 - 19, 2008, Kanazawa, Japan

[14] Y-Ch. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. Accuracy characterization for metropolitan-scale Wi-Fi localization. In *Proeedings of the 3rd int conference on Mobile systems, applications, and services* (MobiSys '05). ACM, NY, USA, pp. 233-245.

[15] S. Consolvo et al. Conducting In Situ Evaluations for and with Ubiquitous Computing Technologies. *International Journal of Human-Computer Interaction*, 22(1), pp. 107-122, 2007

[16] A. Coronato, M. Esposito, G. Pietro: A multimodal semantic location service for intelligent environments: an application for smart hospitals. *Personal and Ubiquitous Computing* 13(7), pp.527–538

[17] D. Dearman, A. Varshavsky, E. De Lara, K.N. Truong. An exploration of location error estimation. In *Proceedings of the 9th international conference on Ubiquitous computing* (UbiComp '07). Springer-Verlag, Berlin, Heidelberg, 181-198

[18] R. Hansen, R. Wind, C.S. Jensen, and B. Thomsen, "Seamless indoor/outdoor positioning handover for location-based services in Streamspin", in *Proc. of 10th Int. Conf. on Mobile Data Management, MDM 2009,* pp. 267–272, Taipei, Taiwan, May 2009

[19] A. LaMarca et al.. Place Lab: Device Positioning Using Radio Beacons in the Wild. *In Proceedings of Pervasive 2005*, pp. 116–133

[20] H. Lemelson, M.B. Kjærgaard, R. Hansen, T. King. Error Estimation for Indoor 802.11 Location Fingerprinting. *In Proceedings of LoCA 2009*, pp. 138-155.

[21] W. Niu and J. Kay. Location conflict resolution with an ontology. In *Proc. 6th Int Conf on Pervasive Computing*, pages 162–179, 2008

[22] L. Oppermann, G. Broll, M. Capra, S. Benford: Extending authorizing tools for location-aware applications with an infrastructure visualization layer. *Proceedings of the 8th Int Conf. on Ubiquitous computing* (UbiComp '06), pp. 52–68

[23] S.N. Patel, J.A. Kientz, S. Gupta. Studying the Use and Utility of an Indoor Location Tracking System for Non-Experts. *In the Proceedings of the 8th International Conference on Pervasive Computing* (Pervasive 2010). Helsinki, Finland. pp. 228-245

[24] A. Ranganathan, J. Al-Muhtadi, S. Chetan, R. Campbell, and M.D. Mickunas, "MiddleWhere: a middleware for location-awareness in ubiquitous computing applications", in *Proc. of Int. Middleware Conference, Middleware 2004,* LNCS, vol. 3231, pp. 397–416.

[25] D. Reilly, D. Dearman, V. Ha, I. Smith, K. Inkpen. "Need to Know": Examining Information Need In Location Discourse. In *Proc of Pervasive 2006,* pp. 33–49.

[26] G. Stevenson, J. Ye, S. Dobson, P. Nixon. LOC8: A Location Model and Extensible Framework for Programming with Location. *Pervasive Computing* 9(1), pp.28–37, 2010

[27] G. Vanderhulst, K. Luyten, K. Coninx. Middleware for Ubiquitous Service-Oriented Spaces on the Web. *In: Proc. of AINA Workshops 2007*, pp.1001–1006

[28] A. Varshavsky, S. Patel. Location in Ubiquitous Computing. In: *Ubiquitous Computing Fundamentals*, 2009, pp.285–320.