Sharing Sensor Data in Intelligent Environments

Tim Schwartz, Dominik Heckmann, Jörg Baus

Saarland University and DFKI
66041, Saarbrücken, Germany
schwartz@cs.uni-sb.de, heckmann@dfki.de, baus@dfki.de

Abstract. Instrumented environments are enriched with sensors,senders and computing devices in order to support intelligent applications for the human-computer interaction. Because the sensors and senders in the environment can deliver a large amount of data, these so-called "intelligent environments" form ideal playgrounds to test the novel idea of *ubiquitous knowledge discovery*. In this paper we will describe the sensor architecture and the management software that is installed in the SUPIE (Saarland University Pervasive Intelligent Environment) for sharing basic sensor data. As an example application for ubiquitous knowledge discovery we describe our positioning system that collects data from different types of senders and that derives the user's position by fusing all data that can be helpful for this task.

1 Introduction and Architecture

Before we start representing our ideas about knowledge discovery in instrumented environments, we will briefly introduce the Saarland University Pervasive Instrumented Environment (SUPIE), in which computational resources are embedded as well as distributed. The environment's hardware and software architecture has been designed for the seamless integration of various services and applications supporting different tasks such as our shopping assistant [1] and the pedestrian navigation system [2]. The software architecture consists of four hierarchical layers, see figure 1 and [3], where assistance applications are considered as the top level.

The actual assistance applications of our intelligent environment use the knowledge representation and services of the lower layers to provide an intelligent user interface. The shopping assistant application provides product information and personalized advertisements to the user, this also includes the animated agent [4]. As the user interacts with real products on the shelf, their actions are recognized by a RFID reader and sent as events to the application. In response, the assistant proactively serves product information to the user, either on a tablet or a wall mounted plasma display. The user can also use their PDA for multi-modal interaction with the shopping assistance application, which entails the fusion of speech, handwriting, intra and extra gestures. Our navigation application also runs on a PDA and is based on the information provided by the location model and the positioning service. On the handheld, a graphical map and speech synthesis are provided. Besides the mobile device, the system utilizes nearby public displays to present arrows that indicate the direction to go.

All these aforementioned applications have access to a knowledge representation layer. This layer models some parts of the real world like an office, a shop, a museum or an airport, see [5]. It represents persons, things and locations as well as times, events and their properties and features. A hierarchical symbolic location model represents places at different levels of granularity, like cities, buildings and rooms, and serves as a spatial index to the situational context. In order to generate localized presentations and navigational aid, the positions of the user, the buildings and the displays have to be known. Therefore the symbolic world model is supplemented by a geometric location model, containing the necessary data.

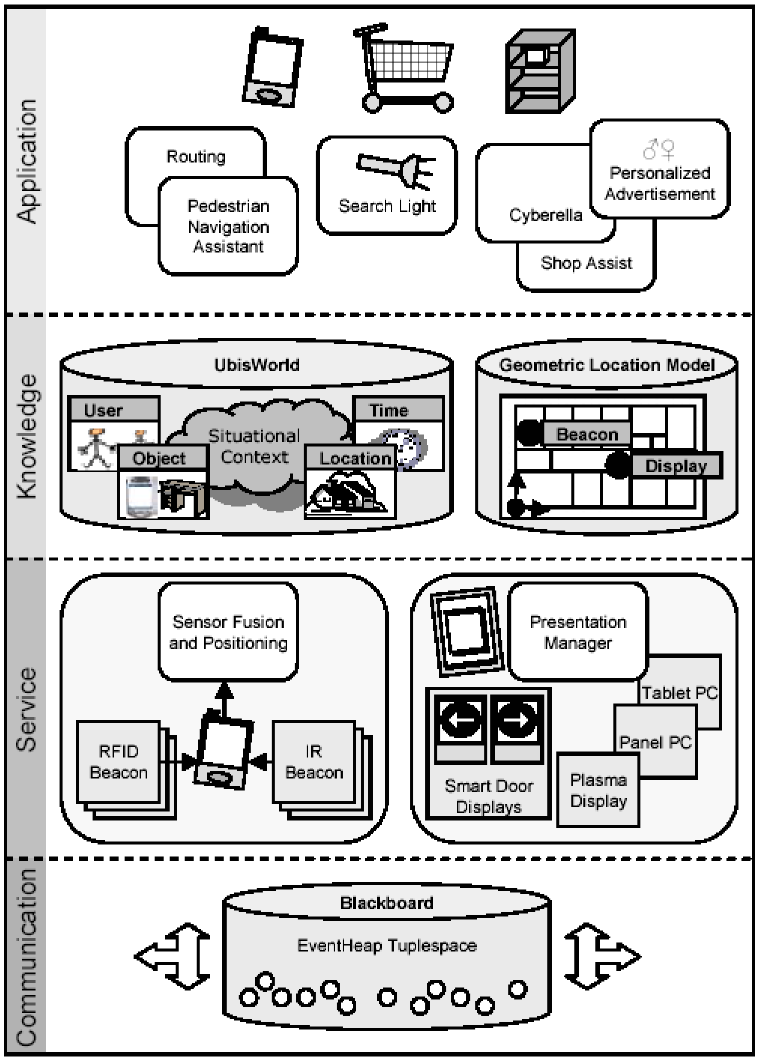

Fig. 1. The four-layered architecture of instrumented environments with communication, service, knowledge and application

Our software architecture's service layer provides multiple applications at the same time with information about a user's position in the environment and offers access to the public presentation service. It hides the technical complexity of these services behind a simple interface, which is based on blackboard events. For the positioning service we adopt a heterogeneous sensor approach, where a mobile terminal receives coordinates from infrared beacons as well as active RFID tags and estimates its own position using a dynamic Bayesian networks approach. The positioning service is presented in more detail later in the article as example for gathering sensor data. The presentation service provides a simple interface that allows applications to present Web content such as HTML and Flash on any display, which is connected to the presentation manager.

The communication and coordination within the intelligent environment is based on a commonly accessible tuplespace. Processes post their information to the space as tuples (collections of ordered type-value fields) or read them from the space in either a destructive or non-destructive manner. As the backbone of our communication layer we have chosen the EventHeap server and API, developed at Stanford University as a framework for their iRoom project (see [6]). Similar implementations are available from Sun [7] and IBM [8].

1.1 Our Notion of Ubiquity

The presented approach can be classified as *ubiquitous computing*. Mark Weiser's classification of a ubiquitous computing system is based on two fundamental attributes: namely *ubiquity* and *transparency*, see [9]. Ubiquity denotes that the interaction with the system is available wherever the user needs it. Transparency denotes that the system is non-intrusive and is integrated into the everyday environment. Further inspiring statements by Weiser are:

– Ubiquitous computing is fundamentally characterized by the connection of things in the world with computation.
– The main idea of ubiquitous computing: integrate computing into objects of daily life but hide its existence if possible.
– Things in the world can be actively supported by integrating computing devices or adding additional identification badges or labels. Things can be connected into Intelligent Environments via e.g. wireless lan.
– The real power of the concept comes not from any one of these devices; it emerges from the interaction of all of them. The hundreds of processors and displays are not a "user interface" like a mouse and windows, just a pleasant and effective "place" to get things done.

2 Sensors

There are different kinds of sensors integrated in our intelligent environment ranging from smart sensors boards which are able to sense, e.g. lighting conditions, temperature or physical interactions with objects, bio-sensors (see [10]), to passive/active RFID-Tags and Bluetooth-Dongles, which are used for our positioning services described in the next section.

Instead of describing all these sensors in a shallow fashion, we put the focus on the sensors for location and describe them in detail in the following.

The knowledge about the user's position is valuable information in a variety of applications. Because the Global Positioning System (GPS) that is normally used for such purposes is not available in buildings a different technology has to be used. One idea is to equip the user with a sender and to instrument the building with respective sensors. These sensors detect the signal the user's sender is dispatching and send this information to a centralized server that can then calculate the position of the user. This kind of localization is often called tracking or *exocentric localization* ([11]) because the user is sharing her position with the environment (the sender shouts "I'm here, I'm here!"). The opposite approach is to place the senders in the building and to let the user wear a device that is equipped with the respective sensor. The senders in the building send a signal ("You're near me, you're near me!"), the user's sensor detects this signal and the personal device of the user calculates the position. This is called positioning or *egocentric localization* because no information is send to the environment and thus the user's privacy is better protected.

Sensors/senders that are often used for the purpose of localization are: Ultrasound, infrared, and various radio based devices like WiFi, RFID and Bluetooth. These technologies differ in cost and reliability, where cheaper senders often provide less accuracy than higher priced ones. One of the problems of a localization system is that they normally use just one kind of sensor/sender technology and such a system works only in those buildings that provide the respective infrastructure. Our idea is to use different sensors/senders and to use a sensor fusion approach to calculate the users position. The advantage of this system is that it can work if there is just one of the sensor/sender technology available or it can derive a better position if more sensors/senders are available. Because such a system tries to always reach the highest possible accuracy (using all of the sensors/senders at the current position) we call it an Always Best Positioned (ABP) system (in analogy to Always Best Connected). We think that such an ABP system is a good example for ubiquitous knowledge discovery.

Our ABP system currently uses active RFID tags and infrared beacons as senders and an RFID reader card and the built-in infrared port of a PDA as sensors (see Figure 2). To fuse the sensory data we use an approach that is based on geo referenced Dynamic Bayesian Networks (see [11, 12] for a detailed description). The system is an egocentric system (as explained above) so the user can decide if she wants to reveal her position to the instrumented environment. She can do so by clicking a corresponding menu entry which causes the positioning engine to send the positioning information on the iROS Event Heap (see [13] for more information about the Event Heap).

Since every room in our lab has as least one computer in it and we have some public displays installed in exposed places, we use an inexpensive but also coarse exocentric (tracking) system that uses standard USB Bluetooth dongles to detect the presence of Bluetooth enabled phones. A small Java program is installed on the public displays and the office computers that scans for bluetooth devices. The result of each scan is also posted on the iROS Event heap as a list of all detected Bluetooth-addresses, the "friendly names" and of course the position of Bluetooth dongle itself. This can be

Fig. 2. iPAQ with attached RFID sensor and built-in infrared sensor (left). Active RFID tag and infrared beacon (right).

considered as raw sensor data because it does not contain the position of the detected devices. An application that wants to use this data has to infer the position out of these lists.

3 Sharing Sensor Data and Context Information

The presented event heap technology proves to be sufficient for sharing the sensor data within the instrumented environment itself. However, for sharing the sensor data and further inferred user model and context information with external systems and applications like the SPECTER system, see [14], new tasks of sharing and privacy handling have to be solved.

In our approach, we link the SUPIE event heap with the UbisWorld[1] situational statement service, see [14]. The concept of sharing with external systems is split up

[1] UbisWorld is based on the new concept of ubiquitous user modeling which means that networked systems constantly track the users behavior at many times and in many ways. See `http://www.ubisworld.org`

within UbisWorld into exchanging and integrating statements about sensor data and context information. The former is realized by a user model and special context server (www.u2m.org) that provides a service-based architecture for distributed storage and retrieval of statements about users and situations.

We developed the RDF-based user model and situation exchange language UserML to enable decentralized systems to communicate over user models as well as situational and contextual factors. The idea is to spread the information among all adaptive systems, either with a mobile device or via ubiquitous networks. UserML statements can be arranged and stored in distributed repositories in XML, RDF or SQL. Each mobile and stationary device has an own repository of situational statements, either local or global, dependent on the network accessability. A mobile device can perfectly be integrated via wireless lan or bluetooth into the intelligent environment, while a stationary device could be isolated without network access. The different applications or agents produce or use UserML statements to represent the user model information. UserML forms the syntactic description in the knowledge exchange process. Each concept like the user model auxiliary *hasProperty* and the user model dimension *timePressure* points to a semantical definition of this concept which is either defined in the general user model ontology GUMO, the UbisWorld ontology, which is specialized for ubiquitous computing, or the general SUMO/MILO ontology. More about these ontologies and the used protocols can be found in [15].

Figure 3 shows the input and output information flows *add*, *request* and *report* of the SituationService. They are denoted as (yellow) arrows. The numbers in the (or-

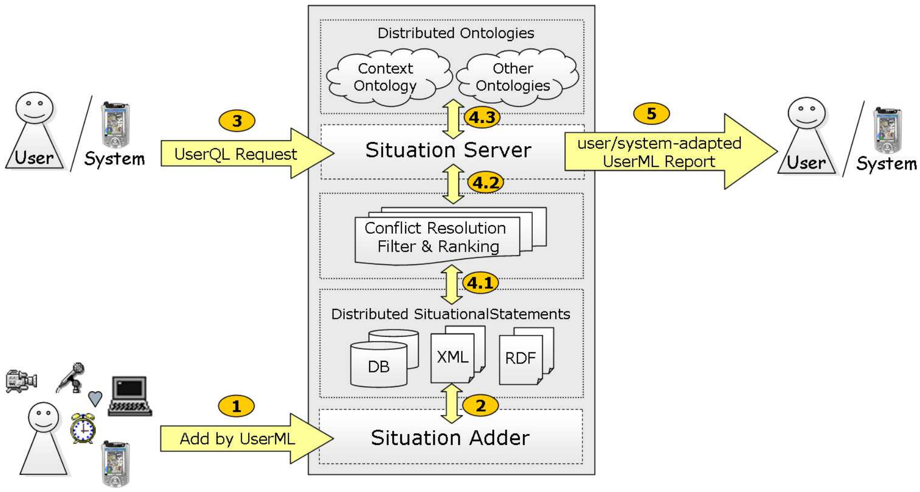

Fig. 3. General procedural view to the SituationService

ange) ovals present the procedural order. Number (1) visualizes the sensors, users and systems that add statements via `UserML`. The statements are sent to the so called *Sit-*

uation Adder, a parser that preprocess the incoming data and distributes them to the different repositories, as indicated by number (2). If now a request is sent to the *Situation Server* via `UserQL` from a user or a system, see number (3), the repositories are selected from which the statements are retrieved as shown at number (4.1). Then conflict resolution strategies are applied, see number (4.2), and the semantic interpretation as indicated by number (4.3). Finally, see number (5), the adapted output is formatted and sent via HTTP in form of an `UserML` report back to the requesting user or system.

The integration of statements is achieved with an accretion model together with a multilevel conflict resolution method [5], which also solves the problem of contradictory information. What statements can be retrieved and how they are integrated depends on several layers of metadata attached to the statements by means of reification. From the outermost to the innermost layer, these are: administration, privacy, explanation, and situation. They establish a sequence of access constraints which have to be met in order to obtain the reified statement. The privacy layer in this sequence is of special interest. It implements the following privacy attributes: key, owner, access, purpose, and retention. The UbisWorld service checks these attributes in order to deliver as much information as possible without violating the users preferences. Combined with the other layers, complex situational access constraints can be established.

4 Summary

We have described the foundation for sharing sensor data within the Saarland University Pervasive Instrumented Environment by event heap technology, while we have directed the focus on location sensors and the diversity within the overall architecture. As an example for ubiquitous knowledge discovery we described how to fuse the data from different sender/sensor-technology to derive the position of a user. This example also shows a method for integrating privacy issues into the aspect of sharing sensor data of instrumented environments with external systems.

Acknowledgments

This research is being supported by the German Science Foundation DFG (Deutsche Forschungsgemeinschaft) in its Collaborative Research Center on Resource-Adaptive Cognitive Processes, SFB 378, Project EM 4, BAIR, and its Transfer Unit on Cognitive Technologies for Real-Life Applications, TFB 53, Project TB 2, RENA.

References

1. Wasinger, R., Schneider, M., Baus, J., Krüger, A.: Multimodal Interaction with an Instrumented Shelf. In: Proceeding of Artificial Intelligence in Mobile Systems 2004 (AIMS 2004). (2004) pages 36–43
2. Krüger, A., Butz, A., Müller, C., Wasinger, R., Steinberg, K., Dirschl, A.: The Connected User Interface: Realizing a Personal Situated Navigation Service. In: Proceedings of the International Conference on Intelligent User Interfaces (IUI 2004), ACM Press (2004) 161–168

3. Stahl, C., Baus, J., Brandherm, B., Schmitz, M., Schwartz, T.: Navigational- and shopping assistance on the basis of user interactions in intelligent environments. In: Proceedings of the IEE International Workshop on Intelligent Environments (IE), University of Essex, Colchester, UK (2005)
4. Kruppa, M., Spassova, L., Schmitz: The Virtual Room Inhabitant. In: Proceedings of the 2nd Workshop on Multi-User and Ubiquitous User Interfaces (MU3I), San Diego (CA). (2005)
5. Heckmann, D. LNAI3946. In: Situation Modeling and Smart Context Retrieval with Semantic Web Technology and Conflict Resolution. Springer-Verlag, Berlin Heidelberg (2006) 34–47
6. Fox, A., Johanson, B., Hanrahan, P., Winograd, T.: Integrating information appliances into an interactive workspace. IEEE Computer Graphics and Applications **20** (2000) 54–65
7. Freeman, E., Hupfer, S., Arnold, K.: JavaSpaces Principles, Patterns and Practice. Addison Wesley (1999)
8. Wyckoff, P.: Tspaces. IBM Systems Journal (1998)
9. Weiser, M.: Some computer science issues in ubiquitous computing. Communications on the ACM **36** (1993) 75–84
10. Brandherm, B., Schultheis, H., von Wilamowitz-Moellendorff, M., Schwartz, T., Schmitz, M.: Using physiological signals in a user-adaptive personal assistant. In: Proceedings of the 11th International Conference on Human-Computer Interaction (HCII-2005), Las Vegas, Nevada, USA (2005)
11. Schwartz, T., Brandherm, B., Heckman, D.: Calculation of the User-Direction in an Always Best Positioned Mobile Localization System . In: Artificial Intelligence in Mobile Systems (AIMS) 2005, in adjunction with MobileHCI 2005, Salzburg, Vienna. (2005)
12. Brandherm, B., Schwartz, T.: Geo Referenced Dynamic Bayesian Networks for User Positioning on Mobile Systems. In Strang, T., Linnhoff-Popien, C., eds.: Proceedings of the International Workshop on Location- and Context-Awareness (LoCA), LNCS 3479, Munich, Germany, Springer-Verlag Berlin Heidelberg (2005) 223–234
13. Borchers, J., Ringel, M., Tyler, J., Fox, A.: Stanford interactive workspaces: A framework for physical and graphical user interface prototyping. In: IEEE Personal Communications Special Issue on Smart Homes, June 2002. (2002)
14. Kröner, A., Heckmann, D., Wahlster, W.: SPECTER: Building, exploiting and sharing augmented memories. In: Workshop on Knowledge Sharing for Everyday Life (KSEL06), Kyoto, Japan (2006) 1–8
15. Heckmann, D.: Ubiquitous User Modeling. PhD thesis, Department of Computer Science, Saarland University, Germany (2005)