On the Utilization of Heterogeneous Sensors and System Adaptability for Opportunistic Activity and Context Recognition

Marc Kurz, Gerold Hölzl, Alois Ferscha
Johannes Kepler University Linz
Institute for Pervasive Computing
Linz, Austria
{kurz, hoelzl, ferscha}@pervasive.jku.at

***Abstract*—Opportunistic activity and context recognition systems draw from the characteristic to utilize sensors as they happen to be available instead of predefining a fixed sensing infrastructure at design time of the system. Thus, the kinds and modalities of sensors are not predefined. Sensors of different types and working characteristics shall be used equally if the delivered environmental quantity is useful for executing a recognition task. This heterogeneity in the sensing infrastructure and the lack of a defined sensor infrastructure motivates the utilization of sensor abstractions and sensor self-descriptions for identifying and configuring sensors according to recognition tasks. This paper describes how sensors of different kinds can be accessed in a common way, and how they can be utilized at runtime by using their semantic self-descriptions. The different steps within the lifecycle of sensor descriptions are described to understand the powerful concepts of self-describing sensors and sensor abstractions. Furthermore, a prototypical framework realizing the vision of opportunistic activity recognition is presented together with a discussion of subsequent steps to adapt the system to different application domains.**

***Keywords*—Activity recognition; system adaption; opportunistic activity recognition; heterogeneous sensors**

I. Introduction

Common and established activity and context recognition systems usually define the recognition task together with the sensing infrastructure (i.e., the sensors, their positions and locations, spatial and proximity relationships, sampling rates, etc.) initially, at design time of the system. The successful recognition of activities and more generally the context of subjects is heavily dependent on the reliability of the sensing infrastructure over a certain amount of time, which is often difficult to achieve, due to sensor displacements or sensor disconnects (e.g., a sensor may run out of power). In contrast to that, opportunistic systems utilize sensor systems as they happen to be available to execute a dynamically defined recognition goal [1][2]. The challenge altered from deploying application specific sensor systems for a fixed recognition task to the utilization of sensors that happen to be available for dynamically stated recognition goals [1][3][4]. The available sensor systems have to be discovered, identified, and configured to cooperative sensor *ensembles* that are best suited to execute a certain recognition goal in a specific application domain. Furthermore, an opportunistic system has to be robust and flexible against spontaneous changes in the surrounding sensor environment, allowing the continuity of the recognition process even if sensors disappear (or appear) in the sensing infrastructure [5]. Therefore, three crucial challenges (amongst others) can be identified: (i) the utilization of sensor systems of different kinds and modalities as data delivering entities, (ii) the identification of sensors and their capabilities for configuring ensembles according to recognition goals, and (iii) the adaptation of an opportunistic activity recognition system (together with the sensor representations and the low-level algorithmic dependencies) to a specific application domain. This paper presents the concepts of sensor abstractions [1][6] and sensor self-descriptions [1] to cope with these challenging aspects. Furthermore, a reference implementation of an opportunistic activity and context recognition system is presented, referred to as the *OPPORTUNITY Framework* [1][6][7] accompanied with a discussion how the framework together with the sensor representations (composed of abstractions and self-descriptions) can be easily adapted to diverse application domains.

The remainder of the paper is structured as follows. Section II motivates and presents the concept of sensor abstractions, which enables a common usage of different sensor systems. Section III describes how sensor systems can be utilized and configured dynamically according to an actual recognition goal by using their self-description, and how the sensor self-description evolves over time by illustrating the self-description life-cycle. Section IV discusses how the OPPORTUNITY Framework can be adapted to different application domains. The final Section V closes with a conclusion and summarizes the core contributions of this paper.

II. Sensors in the OPPORTUNITY Framework

Opportunistic activity and context recognition systems do not predefine their sensing infrastructure initially, as it was the usual case in decades of related systems (e.g., *Salber et al.* [8], *Bao and Intille* [9], *Ravi et al.* [10], *Tapia et al.* [11], and *Ward et al.* [12]). Instead, the system makes best use of the currently available sensors for executing a recognition goal. This aspect also includes heterogeneity within the sensing infrastructure, as the lack of a defined sensing infrastructure also includes missing definitions of the kinds and modalities of the sensors involved in an ensemble. Therefore, an activity recognition system that operates in an opportunistic way has to be capable of handling different sources of environmental data.

 ISBN: 978-1-61208-274-5

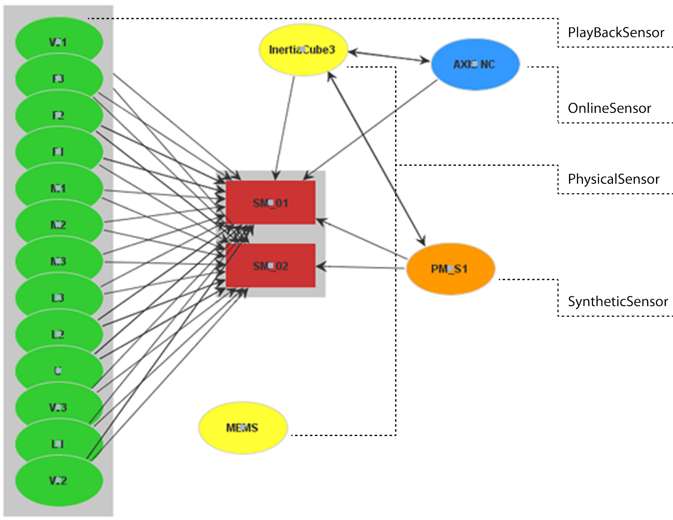

Fig. 1. An actual sensing infrastructure showing different types of available sensors.

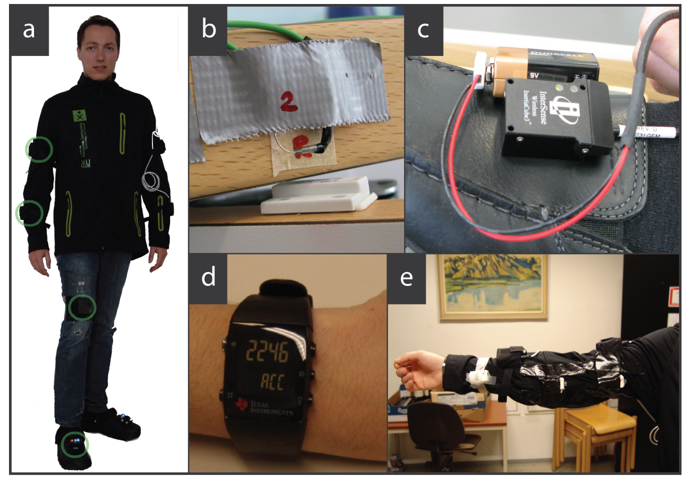

Fig. 2. Impressions of the (physical) sensor systems that are available within the OPPORTUNITY Framework.

These sources do not have to necessarily be physical sensors (e.g., acceleration, orientation, temperature, etc.), but can also be immaterial devices that can provide valuable information to a system [6]. Sensor abstractions [1][6] provide a common and easy accessible interface to handle different kinds of material and immaterial devices as general type *Sensor* (e.g., physical, online, playback, synthetic, and harvest sensors). The abstractions hide the low level access and connection details and provide methods to handle different devices in a common way. This concept enables the inclusion of sensors in ensemble configurations (the set of sensors that is best suited to execute a recognition goal [2][4]) of different kinds, types and modalities as they happen to be available.

The OPPORTUNITY Framework [1][7] is a prototypical implementation (written in Java/OSGi) of a system that recognizes human activities in an opportunistic way. By enabling the utilization of sensors of different modalities, thus does not restrict the sensing infrastructure to be composed of a predefined set of specific sensors, the system is flexible towards the generation of ensembles for activity recognition. By further utilizing the concept of self-describing sensors (see Section III) the system is robust against changes in the sensing infrastructure, thus can react on spontaneous changes on the sensors' availability by reconfiguring the corresponding activity recognition chains and the ensemble [5]. Furthermore, since also immaterial devices like a *PlayBackSensor* [6] - that replays a pre-recorded data source, thus simulates an actual sensor - can be utilized at runtime of the system, this allows the configuration of hybrid simulation scenarios made of physical and simulated (playback) devices. The different classes that implement the hardware access (in case of *PhysicalSensors*), the connection to a remote data source (*OnlineSensor*), or the reading of datasource for *PlayBackSensors* are all derived from a common interface. This means from the framework's point of view all these devices and sources of environmental data can be accessed and utilized in a common way.

Figure 1 displays an example for an actual sensing infrastructure with two active recognition goals (the two red rectangles) within the OPPORTUNITY Framework. This schematic illustration is available as visualization in the OPPORTUNITY Framework and presents the current available sensor devices, the active sensing mission, and the active data flows between the involved units. The entire actual sensing infrastructure in this example consists of 17 sensors, each illustrated by a colored ellipse, whereas 13 are of type *PlaybackSensor* (green), 2 are of type *PhysicalSensor* (yellow), respectively one of type *OnlineSensor* (blue), and one of type *SyntheticSensor* (orange). The arrows in the figure indicate the dataflows from sensors to active recognition goals, and between sensors themselves. Thus, an ensemble is the best configurable set of sensors that cooperates to execute a recognition goal, whereas different types of sensors can be utilized by accessing them in a common, standardized way by providing interfaces and APIs to hide the low-level access details. The following Table I provides an overview of the currently available sensor abstractions in the OPPORTUNITY Framework.

The data sources for the sensors of type *PlaybackSensor* in Table I have been recorded in two recording sessions. First, a kitchen scenario in May 2010 was set up, where 72 sensors with more than 10 modalities have been utilized, and 12 subjects performed early morning activities, each 6 runs. Second, another kitchen equipped with sensors in December 2011, where 5 subjects performed activities like *coffee preparation*, *coffee drinking*, and *table cleaning*. These recording sessions are described in detail in [13] and [3]. Figure 2 provides impressions of the sensors that are made available as *PlaybackSensor* or *PhysicalSensors* in the OPPORTUNITY Framework. This means, they can be replayed anytime and behave as if they would be physically present. This enables the configuration of hybrid and powerful simulation scenarios for opportunistic activity recognition.

Figure 2(a) shows the *MotionJacket* sensor [14], which contains five XSens MTx units, mounted on the upper and lower arms, and one on the upper back position. Furthermore, one bluetooth accelerometer is mounted on the knee of the person, and one SunSPOT device is attached on the shoe toe-box. Figure 2(b) displays a reed switch as it was used in the dataset recording session in [13]. These magnetic switches were mounted in the environment, on different fitments and

TABLE I. OVERVIEW OF CURRENTLY AVAILABLE SENSOR ABSTRACTIONS IN THE OPPORTUNITY FRAMEWORK.

Short Name	*Sensor Type*	*# of Sensors*	*Further Details*
Reed Switch	PlaybackSensor	13	HAMLIN MITI-3V1 Magnetic Reed Switch
USB Accel	PlaybackSensor	8	USB ADXL330 3-axis Accelerometer
BT Accel	PlaybackSensor	1	Bluetooth ADXL330 3-axis Accelerometer
Ubisense	PlaybackSensor	1	UBISENSE Location Tracking System
Shoetoebox	PlaybackSensor	2	Sun SPOT LIS3L02AQ Accelerometer
Motionjacket	PlaybackSensor	5	XSENS Xbus Kit MTx
Motionjacket	PhysicalSensor	n	XSENS Xbus Kit MTx
Ubisense	PhysicalSensor	1 System (n tags)	UBISENSE Location Tracking System
TI Chronos	PhysicalSensor	n	Texas Instruments eZ430 Chronos
SunSpot	PhysicalSensor	n	Sun SPOT LIS3L02AQ Accelerometer
RFID	PhysicalSensor	n	Inside Contactless M210-2G
MEMS Microphone	PhysicalSensor	n	—
IPhone4	PhysicalSensor	n	Iphone4 Sensor Platform
IntertiaCube3	PhysicalSensor	n	InterSense Wireless InertiaCube3
TI EZ430	PhysicalSensor	n	Texas Instruments EZ430 Chronos
AxisCamera	OnlineSensor	n	AXIS 2120 Network Camera
FSA Pressure	SyntheticSensor	n	XSENSOR PX100:26.64.01

household appliances (e.g., drawers, fridges, doors, etc.). Figure 2(c) shows an InterSense Wireless Inertiacube3 capable of 3 DOF tracking (acceleration, gyro, and magnetometer), mounted on the shoes of persons. Clipping (d) of Figure 2 contains an off-the-shelf wrist worn device (i.e., the Texas Instruments EZ430 Chronos) in a watch-like form, that provides acceleration at a maximum sampling rate of 100Hz. The last clipping (e) shows multiple sensors as used in [13] and [3], and as made available in the OPPORTUNITY Framework as sensor abstraction. First, two of the XSens MTx systems (i.e., the *MotionJacket*) mounted on the upper and lower right arm are visible. Second, three of the bluetooth acceleration (the white devices) are shown. These self-constructed devices contain a simple acceleration sensor, a bluetooth communication unit and power supply.

The OPPORTUNITY Framework is meant to be open-ended. This means on the one hand that the abstraction concept is not restricted to the yet identified six abstractions (i.e., *PhysicalSensor*, *PlaybackSensor*, *OnlineSensor*, *SyntheticSensor*, *HarvestSensor*, and *ProxySensor*) [6]. Furthermore, the available sensors and sensor abstractions as presented in Table I and Figure 2 are a starting point in the OPPORTUNITY Framework and subject to add further (abstracted) sensors on demand. The following Section III describes the second important concept in opportunistic systems on the sensor level: *sensor self-descriptions*.

III. UTILIZING SENSORS

One major research challenge in an opportunistic activity recognition system is the fact that the sensor devices are not known at design time of the system. This means the system has to be able to handle devices of different modalities and kinds, and has to react on spontaneous changes in the sensing infrastructure. For enabling an opportunistic system to handle and access a possible variety of different devices and modalities - even material and immaterial devices - we discussed the concept of *Sensor Abstractions* in the previous Section II. Since not only the sensor infrastructure is subject to changes over time, but also the recognition goal is not defined in an opportunistic system, thus can be stated by users or applications at runtime [1][2][4], the set of sensors has to be identified that can be utilized for a recognition goal. This means that each sensor needs a description on a semantic level that provides the information to the system what it can be used for, how it has to be configured (e.g., which sensor signal features and classification algorithms have to be used, which parameters are required, etc.), and what is the expected performance. Therefore we propose the concept of *Sensor Self-Descriptions* providing information what the sensor can be used for and how it has to be configured [1].

The sensor self-description - as the name already tells - describes a sensor, thus provides relevant information about the physical and working characteristics and the recognition capabilities to the opportunistic activity and context recognition system. The description itself is tightly coupled to a sensor and has to meet different requirements, like (i) *machine-readability*, (ii) *human-readability*, (iii) *ease of access*, and (iv) *extensibility*. Taken these requirements, the decision about the format for the sensor self-descriptions is obvious: XML, respectively SensorML [15]. This XML language specification provides standard models, schemes and definitions for describing sensors and measurement processes.

The self-description of sensors is designed to semantically describe the sensing device on a meta-level regarding its working and physical characteristics (e.g., dimensions, weight, power consumption, sampling rate, etc.), and its recognition capabilities and assignment in sensor ensembles for specific recognition goals. These two use cases of the sensor self-descriptions emerge the need to segment them into one part of the description that holds the technical details as they are defined in the corresponding fact sheet delivered by the manufacturer, and into a second part that enables the dynamic configuration of the sensor in cooperative ensembles that aim at executing a recognition goal as accurate as possible. The dynamic part of the sensor self-descriptions contains so-called *ExperienceItems* (Figure 3 shows the important parts from an exemplary ExperienceItem, like the required classifier, the modality of the sensor, the location of the sensor, the recognizable activities together with the DoF value, and the required feature extraction method) [1][16].

Each ExperienceItem acts as snapshot to memorize the sensor capabilities in form of recognizable activities and further information about the sensor (e.g., location, orientation, topology of the place, etc.), thus describes a complete recognition chain [17] (i.e., *data preprocessing and segmentation*, *feature extraction*, *classification*, and *decision fusion*) and the specific methods. Each ExperienceItem features a correspond-

 ISBN: 978-1-61208-274-5

```
...
<characteristics name="ExperienceItem">
 <swe:DataRecord>
  ...
  <swe:field name="method">
   <swe:Text>
    <swe:value>NCC_Classifier</swe:value>
   </swe:Text>
  </swe:field>
  <swe:field name="sensors">
  ...
  <swe:field name="type">
    <swe:Text>
      <swe:value>ACC</swe:value>
    </swe:Text>
  </swe:field>
  <swe:field name="location">
    <swe:DataRecord>
      <swe:field name="Object">
        <swe:Text>
          <swe:value>Person</swe:value>
        </swe:Text>
      </swe:field>
      <swe:field name="Position">
        <swe:Text>
      <swe:value>BAC</swe:value>
  ...
  <swe:field name="labellist">
   <swe:DataRecord>
    <swe:field name="WALK">
     <swe:Text definition="dof">
      <swe:value>0.75</swe:value>
  ...
  <swe:field name="requiredFeature">
   <swe:Text>
    <swe:value>FX_LocomotionAccFeatures</swe:value>
   </swe:Text>
  </swe:field>
 </swe:DataRecord>
</characteristics>
...
```

Classifier Name

Sensor Modality (Acc = Acceleration)

Location & Position of Sensor

Recognizable Activity incl. expected DoF

required Feature Vector

Fig. 3. Selected parts of an exemplary ExperienceItem as part of the sensor self-description [16].

ing *Degree of Fulfillment (DoF)*, which is a quality-of-service metric in the range of [0, 1], which expresses how well a certain activity is recognized (i.e., the *DoF* is an estimate of the expected accuracy) [1]. The ExperienceItem is used by the framework to configure an available sensor with the required machine learning algorithms and the correct training data (i.e., the complete activity recognition chain) to recognize a certain set of activities. ExperienceItems can either be generated offline by a human expert, or autonomously by the system at runtime. The manual generation of ExperienceItems requires offline labeling and training to gather a classifier model and the translation of the configured algorithms into SensorML, respectively self-description syntax. The more interesting way of generating ExperienceItems is done autonomously by the system by applying transfer learning (a sensor "learns" how to recognize certain activities from other sensors, experience is transferred to enhance the system's overall recognition capabilities) [14].

The segmented sensor self-description has different stages that can be described by the corresponding sensor lifecycle. Figure 4 shows the lifecycle for an exemplary sensor together with the stages and their transitions (i.e., (i) *sensor manufactured*, (ii) *sensor enrolled*, (iii) *expert knowledge*, (iv) *sensor active*, (v) *sensor ready*, and (vi) *sensor out of service*). The lifecycle-stages of the sensor and its self-description are described in the following list:

(i) *Sensor manufactured*: the sensor is ready to use and delivered with its technical specification. In Figure 4, the example on the left hand side shows an InterSense InertiaCube3 sensor with the corresponding datasheet. Neither the technical self-description, nor the dynamic description (in SensorML [15] syntax, as required in an opportunistic activity and context recognition system) is yet available at this stage in the lifecycle. The base for specifying and generating the technical self-description is given with the datasheet delivered with the device by the manufacturer.

(ii) *Sensor enrolled*: this stage in the sensor lifecycle occurs once the technical sensor self-description is available. This means the sensor is ready to be used within an opportunistic activity recognition system but still has no ExperienceItems in its dynamic description that enable the involvement in the execution of recognition goals.

(iii) *Expert knowledge*: this stage can be seen as extension to the previous stage (sensor enrolled). A human expert, who manually adds ExperienceItems to the dynamic sensor self-description, can extend the available (dynamic) self-description. This involves offline training and the manual extension of the dynamic sensor self-description by adding ExperienceItems.

(iv) *Sensor active*: The sensor is active, which means it is involved in the process of executing a recognition goal. The role of the sensor can either be that it is integrated in a running ensemble, or that it is involved as learner. That means its sensor self-description is extended autonomously by the system, by adding further ExperienceItems by observing the configured ensemble and its recognition results.

(v) *Sensor ready*: The sensor is ready to be used within the execution of specific recognition goals but is not currently involved in a running ensemble. That means its self-description already contains one or more ExperienceItems. In this passive mode, the enhancement of the self-description can be done again by a human expert in an offline way.

(vi) *Sensor out of service*: The sensor is outdated, which can be the case once a newer version of a specific sensor type is available. The corresponding self-description is versioned and made available for future use with the newer sensor device. The technical description might be outdated but the gathered experience in the dynamic sensor self-description could be of high value for the new device in future recognition goals.

The combination of the two concepts on the sensor level (i.e., sensor abstractions and sensor self-descriptions) represents a data delivering entity in an opportunistic activity recognition system. The step towards a whole new paradigm in

 ISBN: 978-1-61208-274-5

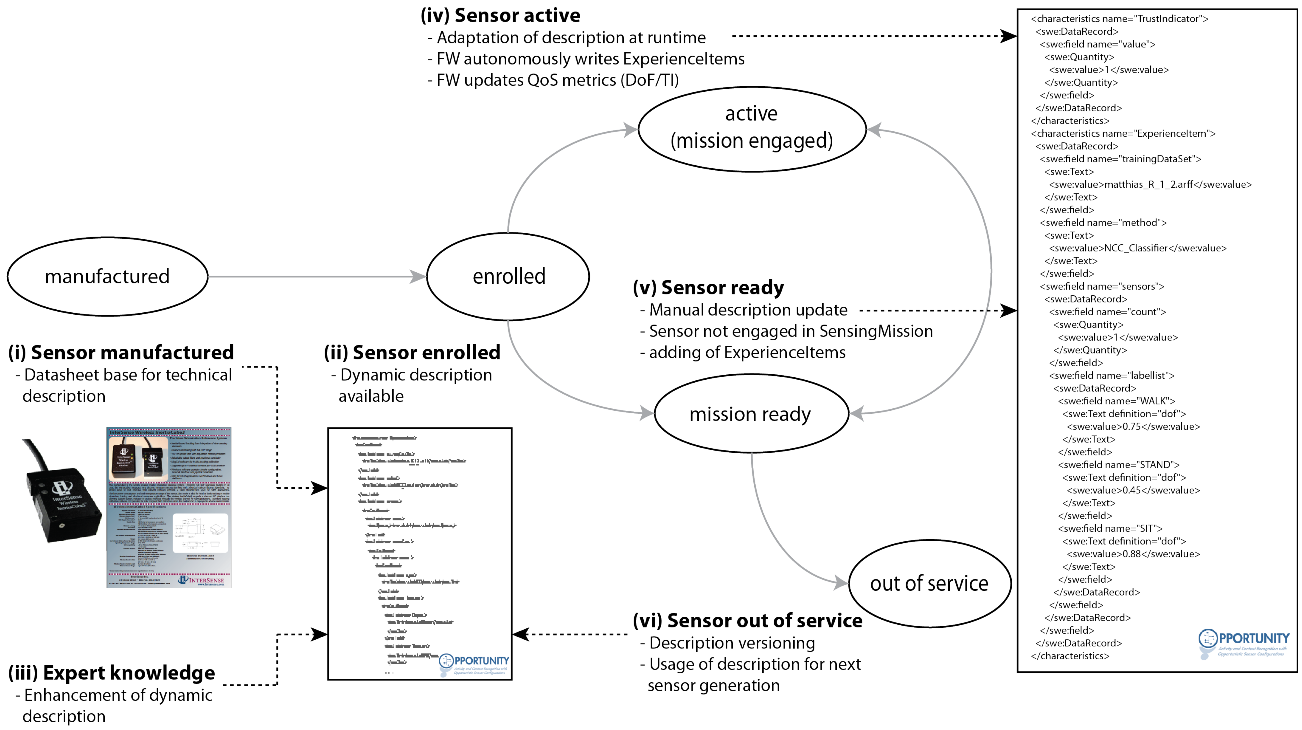

Fig. 4. The lifecycle of sensors and the corresponding self-descriptions in the OPPORTUNITY Framework.

activity recognition without the application dependent deployment of sensors and the reliability of the sensing infrastructure is done and opens new challenges (and possibilities) in open-ended systems. The foundation for flexible and robust activity recognition in an opportunistic way is given, the following Section IV discusses necessary steps to adapt the OPPORTUNITY Framework (and the accompanying sensor representations) to different application domains.

IV. Framework Adaptation

The OPPORTUNITY Framework together with the machine learning technologies, the sensor representations and high-level goal processing concepts has to be adaptable to different domains with less possible efforts. The developed concepts operate application and domain independent, they have to be taken and adapted accordingly. Based on the available reference implementation or an already existing domain- or application specific adapted release of the framework, the adaption process itself consists of at most three independent steps:

(i) Activity knowledge extension or replacement.

(ii) Sensor system inclusion to enhance the set of possible and accessible sensors.

(iii) Extension of the sensors self-descriptions.

An example for a necessary framework adaptation is the deployment of the system in private households for activity recognition in order to implicitly control electronic devices. This adaptation of the OPPORTUNITY Framework for optimized energy consumption in private households is described in detail in [18]. There, a field study has been conducted to evaluate the energy saving potentials based on the inhabitants activities (e.g., if someone is not watching television, the TV set can be safely switched off). The OPPORTUNITY Framework (which was used for activity recognition) has been adapted accordingly to meet the requirements and characteristics in such an application. Sensor abstractions have been added to make the expected sensors (i.e., smart phones, wrist-worn accelerometers, Ubisense positioning sensors) available in the application. New sensor descriptions have been added and existing descriptions have been modified to represent the recognition capabilities based on activity representations and relations (in form of an OWL-ontology). The system was able to (i) run stable over a two-week period in each household and to (ii) handle dynamically varying sensor settings.

In the following, these three steps are described in detail, and Figure 5 presents an illustration of the adaptation-workflow. Either all of the steps are executed or a subset of them, which is shown in the figure, to come from the starting basis of the framework (left-hand side) to the domain-adapted framework (right-hand side), depending on the situation.

A. *Knowledge Representation Modification*

The activity knowledge representation is composed using the W3C standard language OWL. Its purpose is to describe activities, the relations among them and more generally the context for a specific application domain. It is left to the application developer how this knowledge is designed, whether it is for example following the development criteria of a taxonomy (strictly hierarchical), or other semantic structures (e.g., ontologies, topic maps, etc.). In [1] we present an example for an ontology, which provides activities, movable and environmental objects, as network of relations for a kitchen

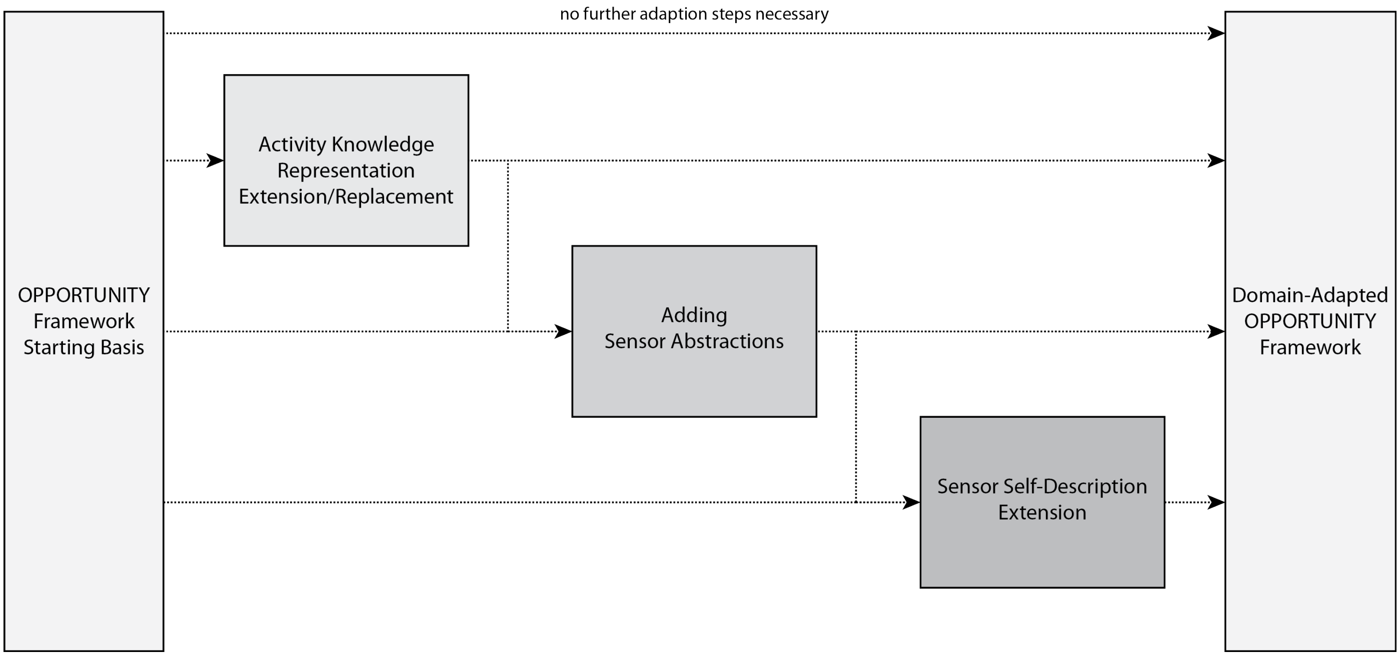

Fig. 5. The application-/domain-specific framework adaptation as three step process.

scenario, containing more than 130 different classes. The ontology itself builds the knowledge base for an application by providing a vocabulary and relations between terms explaining their relationship to each other.

B. Sensor System Inclusion

As already discussed, an opportunistic system does not restrict the kinds and modalities of sensors that act as input sources for environmental quantities. Therefore, to adapt the framework to a new domain, it might be necessary to add sensor abstractions to meet the requirements of possibly occurring sensors. This means, that an application developer, who adapts the framework has to add sensor abstractions by using the defined and common API in form of an interface that acts as common base for having a general way of accessing sensors. Once the abstraction for a sensor device is included in the framework, all appearing sensors of this type can be accessed equally and operate as general type *sensor*. The challenging aspects within the sensor system inclusion step are the low level access details, which have to be implemented once. From the framework's point of view - as all devices are derived from the interface that defines a sensor - those low level details of accessing the device are hidden. Not only material devices (e.g., acceleration, temperature, humidity, orientation sensors) are possible as sources of environmental quantities, but also immaterial sources, like online accessible webservices (e.g., weather or traffic information) can be of high value in an activity and context recognition system.

C. Self-Description Extension

The final step in the framework adaptation work flow is the extension of the sensor self-descriptions. If a completely new sensor type has been added in the previous step as new sensor abstraction, the inclusion of an accompanying new technical description is necessary (see Figure 4). This has to be done only once for each sensor type, since the technical description is static and shared among sensor of the same type. The modification of the dynamic sensor self-description can either make an extension of the existing descriptions and ExperienceItems necessary, or a definition of completely new dynamic descriptions. The first case occurs, whenever existing sensor devices are re-utilized for a new application domain. This makes the extension of the existing dynamic self-descriptions necessary to cover the new activity definitions according to the accompanying ontology by adding new ExperienceItems. The second case occurs, whenever new sensor devices are added and utilized in a new application (domain). This means, new dynamic self-descriptions have to be generated for each device initially. The extension of recognition capabilities in form of ExperienceItems can either be done before operation manually, or during runtime of the system autonomously (as described in [14]).

V. Conclusion and Future Work

This paper presents the two concepts of sensor abstractions and sensor self-descriptions that are big steps towards the vision of recognizing human activities in an opportunistic way (shown in a reference implementation called *OPPORTUNITY Framework*). The capability of utilizing heterogeneous devices by abstracting them to a generalized type - which can be of material and immaterial nature - enables flexible, continuous and dynamic activity recognition with presumably unknown sensor settings. The sensor self-descriptions provide semantic information about individual devices with respect to their capability of recognizing specific activities. This allows for (i) dynamically configuring activity recognition chains at system runtime, and (ii) to react on spontaneous changes in the sensing infrastructure in terms of appearing and disappearing sensor devices. The sensor (self-description) lifecycle and the stepwise adaptation of the OPPORTUNITY Framework to

 ISBN: 978-1-61208-274-5

specific application domains is discussed, whereas this can be broken down to three subsequent steps (i.e., (i) knowledge representation extension, (ii) sensor system inclusion, and (iii) self-description extension). The major contributions of this paper can be summarized to (i) the discussion and the proof of concept of the sensor representation composed of abstractions and self-descriptions, (ii) the identification of a sensor lifecycle representing the sensor's evolution over time, and (iii) - based on the previous items - the stepwise adaptation of an opportunistic activity recognition system to specific application domains.

Future work within the topic of utilizing heterogeneous sensors for accurate activity recognition will tackle the multi-sensor combination with sensor fusion technologies [19] for the specific activity classes. As discussed in related work (e.g., *Kuncheva* and *Whitacker* [20]), the prediction of the accuracy of multi-sensor combinations (i.e., ensembles) is a very challenging task. Currently, research work is conducted that utilizes the mutual information of pairwise sensor combinations in order to predict the accuracy of dynamically configured ensembles. Furthermore, shaping and optimization is currently investigated, meaning that the set of sensors that is included in an ensemble has to be well selected. If a desired activity can be recognized by a lot of sensors, including all of them in the ensemble does not necessarily mean that the accuracy is higher than including only a subset of sensors (the accuracy can even be worse). Therefore, the ensembles have to be optimized towards a maximized expected accuracy for the activities that have to be recognized.

Acknowledgment

The project OPPORTUNITY acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 225938.

References

[1] M. Kurz, G. Hölzl, A. Ferscha, A. Calatroni, D. Roggen, G. Tröster, H. Sagha, R. Chavarriaga, J. del R. Millán, D. Bannach, K. Kunze, and P. Lukowicz, "The opportunity framework and data processing ecosystem for opportunistic activity and context recognition," *International Journal of Sensors, Wireless Communications and Control, Special Issue on Autonomic and Opportunistic Communications*, vol. 1, December 2011.

[2] D. Roggen, K. Förster, A. Calatroni, T. Holleczek, Y. Fang, G. Troester, P. Lukowicz, G. Pirkl, D. Bannach, K. Kunze, A. Ferscha, C. Holzmann, A. Riener, R. Chavarriaga, and J. del R. Millán, "Opportunity: Towards opportunistic activity and context recognition systems," in *Proceedings of the 3rd IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications (AOC 2009)*. Kos, Greece: IEEE CS Press, June 2009.

[3] G. Hölzl, M. Kurz, and A. Ferscha, "Goal oriented opportunistic recognition of high-level composed activities using dynamically configured hidden markov models," in *The 3rd International Conference on Ambient Systems, Networks and Technologies (ANT2012)*, August 2012.

[4] ——, "Goal processing and semantic matchmaking in opportunistic activity and context recognition systems," in *The 9th International Conference on Autonomic and Autonomous Systems (ICAS2013), March 24 - 29, Lisbon, Portugal*, March 2013, p. 7.

[5] M. Kurz, G. Hölzl, and A. Ferscha, "Dynamic adaptation of opportunistic sensor configurations for continuous and accurate activity recognition," in *Fourth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE2012), July 22-27, Nice, France*, July 2012.

[6] M. Kurz and A. Ferscha, "Sensor abstractions for opportunistic activity and context recognition systems," in *5th European Conference on Smart Sensing and Context (EuroSSC 2010), November 14-16, Passau Germany*. Berlin-Heidelberg: Springer LNCS, November 2010, pp. 135–149.

[7] M. Kurz, A. Ferscha, A. Calatroni, D. Roggen, and G. Tröster, "Towards a framework for opportunistic activity and context recognition," in *12th ACM International Conference on Ubiquitous Computing (Ubicomp 2010), Workshop on Context awareness and information processing in opportunistic ubiquitous systems, Copenhagen, Denmark, September 26 - 29, 2010*, September 2010.

[8] D. Salber, A. Dey, and G. Abowd, "The context toolkit: aiding the development of context-enabled applications," in *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*. ACM, 1999, pp. 434–441.

[9] L. Bao and S. Intille, "Activity recognition from user-annotated acceleration data," in *Pervasive Computing*, ser. Lecture Notes in Computer Science, A. Ferscha and F. Mattern, Eds. Springer Berlin / Heidelberg, 2004.

[10] N. Ravi, D. Nikhil, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *In Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence (IAAI)*, 2005, pp. 1541–1546.

[11] E. Tapia, S. Intille, and K. Larson, "Activity recognition in the home using simple and ubiquitous sensors," in *Pervasive Computing*, ser. Lecture Notes in Computer Science, A. Ferscha and F. Mattern, Eds. Springer Berlin / Heidelberg, 2004, pp. 158–175.

[12] J. A. Ward, P. Lukowicz, G. Tröster, and T. E. Starner, "Activity recognition of assembly tasks using body-worn microphones and accelerometers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1553–1567, 2006.

[13] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, M. Creatura, and J. del R. Millán, "Collecting complex activity data sets in highly rich networked sensor environments," in *Proceedings of the Seventh International Conference on Networked Sensing Systems (INSS), Kassel, Germany*. IEEE Computer Society Press, June 2010.

[14] M. Kurz, G. Hölzl, A. Ferscha, A. Calatroni, D. Roggen, and G. Troester, "Real-time transfer and evaluation of activity recognition capabilities in an opportunistic system," in *Third International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE2011), September 25-30, Rome, Italy*, September 2011, pp. 73–78.

[15] M. Botts and A. Robin, "OpenGIS Sensor Model Language (SensorML) Implementation Specification," OGC, Tech. Rep., Jul. 2007.

[16] M. Kurz, G. Hölzl, A. Ferscha, H. Sagha, J. del R. Millán, and R. Chavarriaga, "Dynamic quantification of activity recognition capabilities in opportunistic systems," in *Fourth Conference on Context Awareness for Proactive Systems: CAPS2011, 15-16 May 2011, Budapest, Hungary*, May 2011.

[17] D. Roggen, S. Magnenat, M. Waibel, and G. Troster, "Wearable computing," *Robotics Automation Magazine, IEEE*, vol. 18, no. 2, pp. 83–95, june 2011.

[18] G. Hölzl, M. Kurz, P. Halbmayer, J. Erhart, M. Matscheko, A. Ferscha, S. Eisl, and J. Kaltenleithner, "Locomotion@location: When the rubber hits the road," in *The 9th International Conference on Autonomic Computing (ICAC2012)*, September 2012, p. 5.

[19] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 3, pp. 226–239, 1998.

[20] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine learning*, vol. 51, no. 2, pp. 181–207, 2003.