In: Beigl, M. et al. (Eds.): CONTEXT 2011, LNAI 6967. Berlin : Springer-Verl., 2011. pp. 278-291

Mental Models of Ambient Systems: A Modular Research Framework

Felix Schmitt, Jörg Cassens, Martin Christof Kindsmüller, and Michael Herczeg

Institute for Multimedia and Interactive Systems
Ratzeburger Allee 160, D–23562 Lübeck, Germany
{schmitt,cassens,mck,herczeg}@imis.uni-luebeck.de

Abstract. This paper outlines our current research program in the fields of ambient intelligence and context-aware computing and the tools we are building to accomplish this research program. From a discussion of our conception of mental models in the domain of ambient context-aware computer systems we derive hypotheses which we intend to test empirically. A modular framework for implementing and assessing situation awareness in humans and computers is introduced. We describe the framework's architecture and illustrate its suitability for its intended purpose. Finally, we present an outline of our next steps towards real world application systems for our research.

Keywords: Context awareness, ambient intelligence, mental models

1 Introduction

During the last few decades, computers have taken on an increasingly important role in our lives. Most households do not only own a PC for Internet access and word processing; dedicated gaming consoles or smartphones are also used on a regular basis. Most of these devices are not commonly associated with the word "computer" although they are computing devices in the strongest sense. Today, computational power sees its most widespread use in embedded platforms. Washing machines, refrigerators, television sets, alarm clocks, audio components, cars, cameras, and many more classes of technical artifacts that surround us every day contain a large amount of hard- and software.

In many of these appliances, networking with other devices is either already implemented or can be accomplished with little effort. This allows not only to harvest computational power from already existing sources, but also to access a multitude of environmental sensors and actuators already connected to the embedded network nodes.

For this mesh of embedded computers that has the potential to enhance the usefulness of technical devices, Weiser [1] coined the term *ubiquitous computing*. Your digital photo camera might send the photos you have just taken to your TV set once you put them close to each other. Your MP3 player might offer its songs to your car's audio system once you are seated.

This scenario gets even more attractive when one thinks of linking these input and output channels with knowledge representations and reasoning mechanisms in order to offer novel context adapted services from a combination of all devices' capabilities. This is what we mean when we are talking about *ambient intelligence* and *context aware computing*. Context awareness of digital artifacts or networks of such needs more than massive amounts of sensor input and distributed output channels. First and foremost, it implies a semantically rich representation of the context, i.e. deeper insight into the situation that is unfolding and its meaning to the persons that are present (and maybe even to absent ones).

What we like to accomplish within technical artifacts is a concept that originates in human beings. People are context aware, and they should be able to be so even when interacting with ubiquitous and ambient computer systems of high complexity. Users should be given a chance to understand what their systems' state is, why the state is as it is, and what will likely happen next. Failure to provide this insight will hinder the applicability of ambient intelligent systems and their acceptance among users in real life applications.

2 Research Program

The concept of context awareness is relevant with regard to ambient intelligent computer systems as well as to their users. The former need context awareness to provide their services in a ways that are unobtrusive and adequate to the situation at hand while the latter depend on the same concept in order to interact with the systems in satisfying and confident ways. It is this double meaning of context awareness that caught our interest in this kind of systems. As our research group is inherently interdisciplinary, our interest in the field of ambient, context aware computing is threefold. In this section, we will outline how these three streaks of our work contribute to our view on the topic.

Regarding ambient, intelligent, and context aware systems, we are particularly interested in:

- the *cognitive science perspective*: how users build, refine, change, and discard their mental models of ambient intelligent systems,
- the *artificial intelligence perspective*: how to build context aware systems, and
- the *HCI perspective*: how to foster the construction of adequate mental models in users of ambient intelligent systems.

We found that current theories of mental models have critical shortcomings when applying them to ambient intelligent systems. To target our research interests mentioned above, we chose a twofold approach. We want to explore the concepts of context and awareness in order to understand the situatedness of users' perception of and interaction with ambient intelligent computer systems. The theoretical concepts form the backdrop for our development of real systems with which we can test our hypotheses. Users' difficulties in constructing appropriate mental models of ambient intelligent systems emerge along two dimensions. First, the supply of good affordances to highlight interaction possibilities becomes difficult when the human-computer interface blends with the physical environment and artifacts. Second, intelligent systems might produce seemingly unforeseeable behavior, possibly without direct user interaction.

3 Theoretical Foundations

To illustrate our point and to motivate the development of our awareness framework, we will outline our theoretical understanding of the domain in this section. First, we will define a taxonomy of common conceptions of mental models and derive a pragmatic working definition. Then we will introduce our understanding of situation awareness and context awareness.

3.1 Mental Models: A Taxonomy and a Pragmatic Approach

The term *mental model* does not refer to a single well defined psychological construct. Various research traditions have coined the term in various ways, emphasizing different aspects in accordance with their respective subject. Each of these conceptions has its purpose in a specific field of research, and before we start discussing mental models, we have to make clear, in which sense we are using the term. It will become obvious that some of the available definitions highlight very different aspects of mental models; one single well-defined concept is not sufficient explaining all empirical findings and relevant concepts in our field of work. Therefore we will point out some of these definitions important to our work and derive a pragmatic working definition.

Modern conceptions of mental models found in literature can often be attributed to one of three lines of research: cognitive science, engineering psychology, and supervisory control. We will describe the former two traditions in more detail. It is important to note that the contribution of the latter can be seen, e.g., through cognitive psychologists like Johnson-Laird [2] interpreting Craik [3] or through engineers and engineering psychologists like Endsley et al. [4] relating their concept of situational awareness to the mental model approach in the supervisory control tradition. Thus, it contributes to our understanding as well. After introducing the basic concepts of mental models defined by cognitive science and engineering psychology respectively, we take a step back and derive a pragmatic definition that satisfies the needs of our work.

The Cognitive Science Tradition. From a cognitive science point of view, mental models are the part of the dynamic knowledge representation structures that is mainly dealing with the *where* information. In terms of Baddeley's [5] working memory model, the structure where mental models are represented is called visuo-spatial sketchpad. The "*where system"* is the structure where mental models of spatial and temporal aspects of the situation at hand are represented. This structure is interconnected with the "*what system"* containing propositional facts.

The term “*mental model”* denotes a specific level of knowledge representation. *Schemata* describe the structure and content of generic knowledge. *Production systems* define declarative and procedural knowledge as well as the processes that transfer one into the other. Mental models can be seen as the highest level representation, namely a dynamically updated image of the actual situation, to which it has a homomorphous mapping. Within this tradition, *reasoning* is regarded as the construction of such models. Cognitive Psychologists dealing with human reasoning like Johnson-Laird [2] or dealing with language comprehension like Van Dijk and Kintsch [6] operate close to this conception of the term.

The Engineering Psychology Tradition. Gentner and Stevens [7] as well as Wilson and Rutherford [8] focus on the fields of engineering of technical artifacts, human-machine interaction and human factors. In this area, it is necessary to distinguish several levels of mental models in several groups of stakeholders that are related to one artifact. According to Norman [9] these are:

1. the *user's conceptual model* of the artifact – the representation of the artifact,
2. the *user's mental model* – the internal representation (cf. prev. sect.), and
3. the *designer's conceptual model* of the artifact.

Conceptual models are mental models as well, and the nomenclature is only to distinguish them from the cognitive science style mental models. Note, though, that mental models in the engineering psychology tradition refer to long term memory structures rather than to working memory representations.

To describe the process of a user's construction of his or her (conceptual) mental model, Kindsmüller [10] coined the term *model induction*. This refers to the fact that mental models are progressively constructed from observations, assumptions and pieces of external information by the user herself. The process can only be fostered by inducing these building blocks in the user, since mental models cannot be “uploaded” to the user's mind as a whole. We use this concept when it comes to identifying design rules for ambient systems.

A Pragmatic Approach to Mental Models. As stated above, these three disciplines (cognitive science, engineering psychology, and supervisory control) have quite different conceptions of mental models. Unfortunately it is sometimes unclear which of these conceptions authors are relating the term mental model to. We basically agree with the notion of Rouse and Morris [11] that detailed understanding of the different concepts may help pushing basic science as well as applications further. Moreover, Rouse and Morris propose a common functional definition of mental models that we deem useful for our field of work, namely the design of complex technical systems: *“Mental models are mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions of future system states.”* This definition does not limit the scope of “systems”, so that it generalizes upon all three notions of mental models found in literature. Nevertheless most applications of the concept mental models in designing and introducing systems can benefit from narrowing down the fuzzy term mental model to one of these specific notions, in order to avoid confusion.

Building on the previously laid out conceptions and connotations of mental models, we can state the following with respect to our research:

1. When talking about mental models, we hereafter refer to what engineering psychologists would call the user's conceptual model of a system. This includes the users' abilities
    a. to *describe* the system's working mechanisms in their own, albeit simplifying manner,
    b. to *explain* the interaction with the system, i.e. effective actions with regard to certain interfaces and the system's reaction to those, and
    c. to *formulate expectations* with regard to the system, that means anticipation of the system's behavior in a given context or situation.
2. The users' mental model in the sense of their internal knowledge representation is also of interest, since understanding the circumstances of its construction, modification, or refusal will help us to formulate rules and guidelines for the design of ambient intelligent systems.
3. These rules are then supposed to influence the designers' conceptual models of the systems they construct, but this construction process is not in the focus of our research.

3.2 Situation Awareness and Context Awareness

When talking about awareness, we need to understand what this term actually means. The concept of *situation awareness* originates in human factors research. As Endsley, Bolté, and Jones [4] put it, situation awareness (SA) means *"being aware of what is happening around you and understanding what that information means to you now and in the future"*. This short definition already identifies three so-called "SA-levels" of mental activity that build on top of each other to constitute situation awareness:

1. *perception* of the environment,
2. *comprehension* of the situation at hand, and
3. *projection* into the future.

These three elements show very clearly how closely interconnected situation awareness and mental models are (see our pragmatic approach to mental models in Sect. 3.1).

Apart from *situation awareness*, another related term that can be found in literature is *context awareness*. Although both concepts are closely related, we see differences with regard to the level of semantic interpretation of the situation at hand. Turner [12] has defined *situation* as the entire set of circumstances and *context* as those identifiable features that have predictive power for the behavior of an agent. This correlates with our understanding of the term *situation* denoting a particular combination of circumstances at a given time, whereas *context* relates to a generalized class of situations (cf. Halliday and Hasan [13]). In a hospital, updating a patient's fever chart might happen in the context of the daily ward round which is instantiated by Dr. Miller and Dr. Bower standing at Mrs. Smith's bedside. Abstracting situations into contexts helps people to cognitively master the manifold of situations they encounter all the time: they have mental models of certain contexts, which in turn enable them to retrieve scripts and schemas that propose default interpretations, behavior and expected outcomes.

"Becoming aware of", that is: identifying, a certain context requires a semantically rich representation of the situation at hand. Since semantics of real world situations are quite hard to model computationally, the question arises to which extent we need these semantics in order to provide useful awareness services to the user. We believe that for most tasks usually targeted with contemporary awareness technologies full context awareness is not necessary, but much simpler inferences will suffice. Despite their lack of a semantic representation of the situation, these systems can nevertheless foster situation awareness in their users. The unique properties of ambient systems, which we mentioned in Sect. 1 as well as in [14], however, require more complex semantic models to account for the sheer variety of possible inputs as well as the implicitness of their interactions with users.

From the conception of awareness outlined above, we can derive the following hypotheses about ambient computer systems and their users:

1. Appropriate mental models of a system are necessary prerequisites for the user's context awareness when interacting with the system.
2. Helping users to build an appropriate mental model of situations can improve the users' situation awareness.
3. A system that is supposed to be context aware with regard to certain interactions (even if they are distributed in space and/or time) needs some sort of "mental model", i.e. a semantically rich representation of the situation at hand.

4 Software Framework

Based on our previously laid out concepts of context awareness and mental models, we have been developing a framework for the engineering of context aware systems. Applications built on top of this framework will help us to understand the implications of ambient intelligence on human-computer interaction. We also use the framework to empirically test our theoretical understanding outlined above.

The design of the architecture is based on several principles to connect with this understanding:

- It facilitates prototype-based, feature-driven development processes for rapid deploy-test cycles.
- It allows generating models of context based on current practice in a user-centered development process.
- It allows to use sensors and actuators which connect to users' existing mental models as well as facilitate the generation of new mental models based on metaphors and current practice.

- The system can give enough explanatory feedback so that the user is aware of the system state.

Ultimately, software engineers who want to equip their system with context awareness will be able to implement our *Modular Awareness Construction Kit*'s (MACK) interfaces and install the desired MACK components along with their software. Manufacturers of, e.g., ambient assisted living solutions or interactive guides for museums can thus offer context aware systems with less development effort, just like they would include interfaces to third-party media servers or communication systems.

The MACK framework's architecture is comprised of several functional components that can be modularly combined to provide the required input and output channels as well as domain knowledge and computation thereupon. The architecture also reflects our approach that users' mental models of MACK-based systems are subject to model induction as introduced in Sect. 3.1 and may thus profit from a variety of media types and modalities, i.e. human-computer interfaces.

4.1 Overview

In Fig. 1, the left- and rightmost columns denote the various sensors and actuators, i.e. the system's input and output channels. These need to implement various properties such as the internal XMPP-based protocol. All information is gathered and distributed via the Awareness Hub, which in turn asks the Reasoning Manager's AI engines for interpretation of the information. User data and settings are kept by a separate component for greater modularity and adaptability. Generic sensors and actuators that are not inherently MACK-enabled are driven by a gateway component.

The interfaces between these components are well-defined for interoperability, extensible for future enhancements, and based on standardized open protocols such as TCP/IP and XMPP. Usage of proprietary APIs is kept minimal.

MACK's origin lies in our development of the MATe office awareness system (cf. [14]), of which MACK can be seen as a generalization and abstraction suitable for various types of application systems. Many of the currently implemented peripherals still show MACK's ancestry in that they are primarily targeted at determining and communicating a knowledge worker's interruptibility. The underlying infrastructure and controller logic can be easily used to drive other sensors and actuators though. From the framework perspective, MATe can be seen as an application system which is heavily based upon a MACK infrastructure.

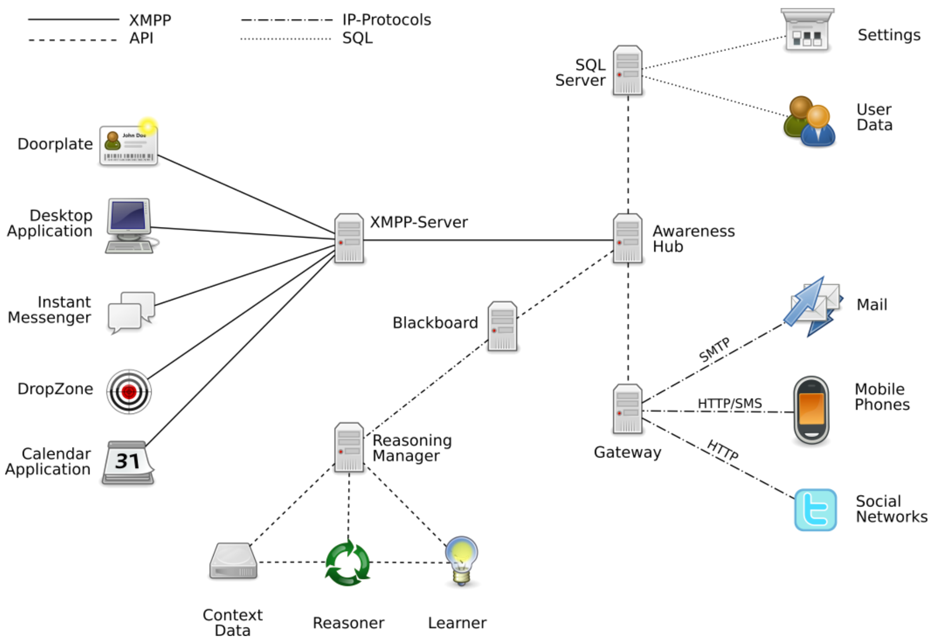

Fig. 1. Architectural overview of the MACK framework

4.2 Sensors

Context-aware systems do not necessarily need to employ a large number of sensors (cf. [15, 16]). On the other hand, we argue that the actual number and types of sensors necessary is depending on the application domain. Our intention to provide a generic framework leads us to create a multitude of sensor modules so that MACK-based application systems can target a multitude of usage scenarios. We will describe two of our peripherals as examples for the inputs our system is able to deal with. The information gathered from some of the sensors does not seem to be valuable by itself. The AwarenessHub will, however, try to relate as many of these inputs as possible to enable the reasoning component to draw semantically rich inferences.

The DropZone is a marked area on the user's desk. People are asked to place a personal token in this area as soon as they enter the room. Despite this explicit interaction, we see the DropZone as an ambient interface, because the tokens are to be carried on the users' key rings. Just like hanging your keys on the key hooks in your hallway when you return home, dropping them in the DropBox area will be highly automatized after only a short period of usage. In our prototype, the token is an optical 2D marker which can be recognized via a camera. This presence information is then forwarded to the AwarenessHub.

With MikeRofone, we aim to identify the number of speakers in a given environment and, via voice signatures taken from individual speech samples, which persons are present. Speech content is not analyzed. The DropZone's camera comes with an integrated microphone which we use to analyze background noise in the room. Due to the large amount of processing power necessary for the analysis, MikeRofone runs on the user's desktop PC.

4.3 Actuators

The ambient character of our system does not only become manifest in its input channels, i.e., sensors, but also with respect to its output channels in the form of various actuators.

One actuator that highlights the ambient nature of our system is the so-called DoorLight. Office doors at our institute have four frosted glass panels of which one can be equipped with LED strips running along two of its edges. A microcontroller evaluates the AwarenessHub's messages and can illuminate the glass panel in either green, yellow, or red color to indicate the user's current interruptibility status. Due to the relatively large area of the "display", co-workers can notice their colleague's status from the corner of their eye while passing down the hallway. For the same reason, the illumination can be dim enough to not draw one's explicit attention.

Another peripheral, the DeskCube, is a combination of actuator and sensor. It consists of a 9x9x9 cm^3 plastic cube, which bears a 8x8 LED matrix on each side. A microcontroller inside the cube uses the matrices to display iconic representations of the six most probable interpretations of the user's current context, the most probable pointing upwards. Thus, the user becomes aware of the system's beliefs. If the system is wrong, the user can simply turn the cube to display the correct context on top. A 3-axis accelerometer within the cube detects the new position and feeds the corrective value back to the AwarenessHub, thereby offering learning input to our reasoners.

4.4 Gateway

The Gateway component allows for the integration of services and devices that are not MACK-enabled. The Gateway's connectivity and functionality can be extended by plug-ins that implement foreign APIs. Currently, this is demonstrated by an SMS service that can be used by the AwarenessHub to notify users of other users' availability or to forward them messages that someone else left on their interactive DoorPlate (cf. [14]).

4.5 AwarenessHub

The AwarenessHub forms the central component of the MACK architecture. It mainly acts as a message passing hub, hence the name. The AwarenessHub keeps track of all other MACK devices including the various peripherals as well as of the user accounts and settings. It forwards messages and retrieves information on behalf of other components while enforcing the users' preference and privacy settings.

The AwarenessHub does not act as a server in the sense that other components log into it to become part of the installation. Instead, the AwarenessHub itself is implemented as an XMPP client. This allows the developers of MACK-enabled application systems to reuse an existing private or even publicly available XMPP server. XMPP supports encrypted connections, so this approach does not in itself imply data protection concerns.

4.6 Reasoning

Since the reasoners are integrated with the rest of the system through a well defined interface they can be easily extended and replaced. For bootstrapping the development, we used only one reasoner for a couple of defined areas of competence each (interruptibility, location of the user, etc.). The reasoners were simple production systems. For empirical testing of our theoretical hypotheses, however, greater flexibility and expressive power is required.

Reasoner. In 2010, Ruge [17] implemented a first reasoner for assessing the interruptibility status of users. This reasoner is based on a domain ontology written in OWL (see below) and the description logics reasoner Pellet [18]. Since this reasoner is based on an open-world assumption, which makes certain inferences impossible, we use an additional close-world reasoner for processing the world model.

For the future, the use of other reasoning paradigms is planned as well, for example using a case-based approach [19]. Compared to other paradigms, case-based reasoning (CBR) suits MACK's needs particularly well for two reasons: First, our system should be able to operate in weak theory domains. The second reason is the relative sparseness of instances of these contexts, i.e. situations that are available to learn from. Both reasons make CBR an especially promising extension of MACK's reasoning subsystem.

Ontology. Ontology development is a highly complex and error-prone task, so it is beneficial to recycle already existing work in the field. Analysis of existing solutions revealed none that would fit our needs, though. Development of Chen et. al.'s [20, 21] *CoBrA-ONT* has been discontinued, and current software tools cannot be used on it. The *CoCa Service Platform* [22] is more up-to-date, but its *Generic Context Management Model* (GCoM) is not freely available. *ONTONYM* [23] provides an extensive model of persons with their respective personal and professional environment, including sensors and properties of technical artifacts. Its structure and handling, however, seemed overly complex for our current state of work, so we opted for an in-house development.

Ruge [17] implemented BOWIE, the *Basic Ontology for Work Information Exchange*. We opted for OWL as the description language, because it is a standard representation for which reasoners exist and which can be used for exchange purposes. In order to work on a unified knowledge model, BOWIE assumes that all information about the world is provided by sensors. Instead of modeling a place's size as an attribute to the `Place` class, a (virtual) `Size` sensor monitors all places and delivers a value upon request. Sub-class axioms and conditional object properties allow for set-based logic operations on the ontology.

Blackboard. MACK's initial reasoning architecture has two main drawbacks. First, several of the different reasoners operate at least partly on the same world model. For example, both the location and the interruptibility reasoner need information on whether the user has placed her personal token in the DropZone (cf. [14]). Second, since the envisioned usage scenarios are in weak theory domains, we deem it beneficial to use different reasoners for the same question and combine their results, for example by using ensemble reasoning [24].

To address these two drawbacks, we are currently restructuring MACK's reasoning subsystem to employ a blackboard architecture [25]. The idea behind blackboards can be understood with the metaphor of a group of human experts who collaboratively solve a problem. The initial problem and world model are written on a blackboard. Whenever one of the experts sees a question he can answer using the known information, he will contribute his results to the blackboard, in the hope that this helps other experts to solve their own tasks.

Our blackboard component uses the same principle: new information from the sensors is written on a shared model in a shared representation language. The different reasoners act upon changed information and write their results to the same model. In our example, the location reasoner could update a user's whereabouts after the DropZone in an office registered the personal token. The updated location information is then used by several interruptibility reasoners who, in turn, update the interruptibility information of the given user.

5 Empirical Evaluation

As we already stated in Sect. 4, MACK is being developed in order to test the viability of our theoretical conceptualizations of mental models and context awareness. From the framework application systems are derived which are deployed among actual end users. One example for the type of systems that can be built on top of the MACK architecture is our office awareness support system MATe. Empirical studies conducted among these application systems' users help us to evaluate to which degree our hypotheses hold in practice. Based on the results we will further refine our theoretical understanding of mental models and context awareness.

Methods used to gain insight into the construction process and structure of users' mental models related to our systems include qualitative interviews and a newly adopted version of the structure-formation-technique by Scheele and Groeben [26] using concept maps (cf. Novak and Cañas [27]) drawn by the users.

Since our software is still considered work in progress, no summative evaluation has been conducted so far. However, since we are using an entangled combination of User-Centered Design (cf. [28]), and Feature-Driven Development [29] we have (bi-) weekly formative evaluations in our development process to make sure that short-coming are detected and repaired early in MATe's/MACK's evolution. These formative evaluations helped us to optimize the common usability of the input and output devices. This is important because we do not want handling problems to interfere with our test results regarding the construction of users' mental models later on.

6 Conclusions and Future Work

In the preceding sections, we have given an introduction to basic concepts that are relevant when dealing with ambient intelligent computer systems. Operating close to the pragmatic definition by Rouse and Morris [11], we conceive mental models as a particular type of long-term memory contents that enable people to describe a system's working mechanisms, to explain their interaction with the system and to anticipate future system behavior. Context is defined as an abstract classification of particular situations, helping people to cognitively master situational challenges by fostering the retrieval of applicable mental models and the knowledge they consist of.

We argue that context awareness (as opposed to situation awareness) requires semantically rich representations and abstractions of the relevant features of a situation at hand. While innate to the human mind, representations of this kind are hard to model computationally. Our *Modular Awareness Construction Kit* (MACK) is intended to implement a software framework for this task. Because of its open architecture, it can be used to construct applications systems in a wide variety of domains.

In the course of our research on ambient, context-aware computing we will use systems built on top of MACK to test our hypotheses which we presented in Sect. 3. For this purpose, though, MACK needs further enhancement with regard to some of its properties and components.

6.1 Reasoning

Most of the existing reasoners are simple production systems, and are linked to the office awareness domain, We plan to both introduce new reasoners suitable in other fields of application and to augment the existing ones with other paradigms.

With more than one reasoner available in the MACK framework, the question arises which of them is right in case they disagree with regard to a question posed by the AwarenessHub. Thus, we need an *Arbiter*. We are currently considering different decision strategies:

- If the different reasoners perform basic ensemble reasoning, its task would be to simply count votes for each outcome and let the majority win.
- If the other reasoners not only deliver their vote, but also an estimate of their own confidence in their decision, it has to calculate the most probably correct decision.

Within MACK's architecture (cf. Fig. 1), the Arbiter will probably not be implemented as another reasoner, but as part of the Reasoning Manager that controls and monitors the various reasoners' access to the blackboard.

6.2 Mobile Client

To leverage more information to reason upon, we will introduce a mobile MACK client for Android-based smartphones. Since users often carry these devices with them, we expect more frequent information updates compared to stationary sensors.

The software will evaluate data from acceleration sensors (walking) as well as location data from GPS and WiFi connections. Furthermore, we can get information on appointments and tasks from other Android apps. As a frontend, the MACK client will allow querying the AwarenessHub for the status of fellow users.

6.3 Evaluation

As soon as the current iteration of code and hardware development is finished, an instance of MACK in the form of the MATe system will be installed on the premises of our research group. In a first step, we will equip several offices with the various components and put the system into everyday use. This will enable us to leverage sufficient log data as well as user experience to conduct a summative evaluation of the system's status quo as well as its ability to support our theoretical understanding of context awareness and users' mental models of ambient computer systems.

References

1. Weiser, M.: The Computer for the Twenty-First Century. Scientific American, 265, 94–100 (1991)
2. Johnson-Laird, P.N.: Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness. Harvard University Press, Cambridge, MA (1983)
3. Craik, K.J.W.: Theory of the human operator in control systems. I. The operator as an engineerig system. British Journal of Psychology, 38, 56–61 (1947)
4. Endsley, M.R., Bolté, B., Jones, D.G.: Designing for Situation Awareness. Taylor & Francis, London (2003)
5. Baddeley, A.D.: The episodic buffer. A new component of working memory? Trends in Cognitive Sciences 4 (11). 418–423 (2000)
6. Van Dijk, T.A., Kintsch, W.: Strategies of discourse comprehension. Academic Press, New York, NY (1983)
7. Gentner, D., Stevens, A.L.: Mental Models. Lawrence Erlbaum Associates, Hillsdale, NJ (1983)
8. Wilson, J.R., Rutherford, A.: Mental models: theory and application in human factors. Human Factors, 31 (6), 617–634 (1989)
9. Norman, D.A.: Some Observations on Mental Models. In: Gentner, D., Stevens, A.L. (eds.) Mental Models, 7–14. Lawrence Erlbaum Associates, Hillsdale, NJ (1983)
10. Kindsmüller, M.C.: Trend-Literacy. Zur Interpretation von Kurvendarstellungen in der Prozessführung. Shaker Verlag, Aachen (2006)
11. Rouse W.B., Morris, N.: On looking into the black box: Prospects and limits in the search for mental models. Psychological Bulletin, 100 (3), 349–363 (1986)
12. Turner, R.: Context-mediated behavior for intelligent agents. In: International Journal of Human-Computer Studies, 48 (3), 307–330 (1998)
13. Halliday, M.A.K., Hasan, R.: Language, Context, and Text: Aspects of Language in a Social-semiotic perspective. Deakin University, Victoria, Australia (1985)
14. Schmitt, F., Cassens, J., Kindsmüller, M.C., Herczeg, M.: Mental Models of Disappearing Systems: Challenges for a Better Understanding. In: Cassens, J., Kofod-Petersen, A., Zacarias, M., Wegener, R. (eds.) Proceedings of the Sixth

International Workshop on Modelling and Reasoning in Context. CEUR-WS.org, Lisbon (2010)

15. Fogarty, J., Hudson, S.E., Lai, J.: Examining the robustness of sensor-based statistical models of human interruptibility. In: Dykstra-Erickson, E., Tscheligi, M. (eds.) CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems, 207–214. ACM, New York, NY (2004)
16. Hudson, S.E., Fogarty, J., Atkeson, C.G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J.C., Yang, J.: Predicting Human Interruptibility with Sensors: A Wizard of Oz Feasibility Study. In: Cockton, G., Korhonen, P. (eds.) CHI '03: Proceedings of the SIGCHI conference on human factors in computing systems, 257–264. ACM, New York, NY (2003)
17. Ruge, L.: Koordinierung von Inferenz- und Lernprozessen für Pervasive-Computing-Anwendungen. Diploma thesis, University of Lübeck (2010)
18. Sirin, E., Parsia, B.: Pellet: An OWL DL reasoner. In: Haarslev, V., Möller, R. (eds.) 2004 International Workshop on Description Logics. 212–213 (2004)
19. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications 7(1): 39-59 (1994)
20. Chen, H.: An Intelligent Broker Architecture for Pervasive Context-Aware Systems. PhD thesis, University of Maryland, Baltimore County (2004)
21. Chen, H., Finin, T., Joshi, A.: The soupa ontology for pervasive computing. In: Calisti, M., Walliser, M., Brantschen, S., Herbstritt, M., Tamma, V., Cranefield, S., Finin, T., Willmott, S. (eds.) Ontologies for Agents: Theory and Experiences, Whitestein Series in Software Agent Technologies and Autonomic Computing. 233–258. Birkhäuser, Basel (2005)
22. Ejigu, D., Scuturici, M., Brunie, L.: CoCA: A Collaborative Context-Aware Service Platform for Pervasive Computing. In: Latifi, S. (ed.) ITNG '07: Proceedings of the International Conference on Information Technology. 297–302, IEEE Computer Society, Washington, DC (2007)
23. Stevenson, G., Knox, S., Dobson, S., Nixon, P.: Ontonym: a collection of upper ontologies for developing pervasive systems. In: Gomez-Perez, J. (ed.) CIAO '09: Proceedings of the 1st Workshop on Context, Information and Ontologies. 1–8. ACM, New York, NY (2009)
24. Plaza, E., Ontañón, S.: Ensemble Case-Based Reasoning: Collaboration Policies for Multiagent Cooperative CBR. In: Aha, D. W., Watson, I. (eds.) Proceedings of the 4th International Conference on Case-Based Reasoning: Case-Based Reasoning Research and Development (ICCBR '01. 437-451. Springer, London (2001)
25. Jagannathan, V., Dodhiawala, R., Baum, L.S. (eds.): Blackboard Architectures and Applications. Academic Press, Orlando, FL (1989)
26. Scheele, B., Groeben, N.: Die Heidelberger Struktur-Lege-Technik (SLT). Eine Dialog-Konsens-Methode zur Erhebung subjektiver Theorien mittlerer Reichweite. Beltz, Weinheim/Basel (1984)
27. Novak, J.D., Cañas, A.J.: The theory underlying concept maps and how to construct and use them. Tech. rep., Florida Institute for Human and Machine Cognition (IHMC) (2008)
28. Norman, D. A. & Draper, S. W.: User Centered System Design: New Perspectives on Human-Computer Interaction. Lawrence Erlbaum, Hillsdale, NJ (1986).
29. Palmer, S. R. & Felsing, J. M.: A Practical Guide to Feature-Driven Development. Prentice Hall, Upper Saddle River, NJ (2002).