Conceptually Advancing “Application Mobility” Towards Design: Applying a Concept-Driven Approach to the Design of Mobile IT for Home Care Service Groups

Dan Johansson, Luleå University of Technology, Sweden

Mikael Wiberg, Uppsala University, Sweden

ABSTRACT

Mobility has become an omnipresent part of our modern IT society. Alongside the general mobility taxonomy of mobile users, terminals, sessions and services, there are also more specialized forms of mobility. Context Awareness Supported Application Mobility (CASAM) or “Application Mobility” is one such form that is explored in this paper. CASAM builds on the idea of using context to move an application between different devices during its execution, in order to provide relevant information and/or services. In this article we use a concept-driven approach to advance mobile systems research, integrating it with a more traditional user-centric method and a case study, further exploring the concept of CASAM. To empirically situate our design work we conducted an empirical study of a home care service group serving the Swedish municipality of Skellefteå, followed by an exercise in matching the properties of the CASAM concept in relation to problems within current workflow (e.g. scheduling, travel, care situation, communication and debriefing). The result is a proposal for an IT artifact manifesting the CASAM concept, attending to all the identified problems while at the same time validating the concept.

Keywords: *Application Mobility, Concept-Driven Design Research, Context Awareness, Health Care Communications, Mobile Technologies*

1. INTRODUCTION

In the mid 90s, Leonard Kleinrock (1996) presented his classic paper entitled “Nomadicity:

Anytime, Anywhere in a Disconnected World”. In his paper, Kleinrock argued that though users were now IT nomads (using IT and computer based services at different places, pausing or shutting down in between), systems were not fully nomadically-enabled. The assumption of us being always connected was wrong. Instead,

DOI: 10.4018/jaci.2012070102

being disconnected was a common mode, and moving from your desk to a conference room in the very same building required a nomadic mode, as the IT environment could be completely different in the two locations.

Indeed, much has happened since Kleinrock wrote his article. Mobile IT usage has become an omnipresent part of our modern society (Beale, 2009; Kaikkonen, 2009; Koblenz, 2009). Recent statistics show that the number of Internet users exceeds two and a half billion, and that there are more than five billion mobile phone subscriptions throughout the world (ITU, 2011). Mobile users act within a space of flows (Castells, 2000), transcending the well-known space of physical places. In the space of flows, both information and technology roam through time and space, more or less unaffected by physical boundaries. Concepts like cloud computing present opportunities to use thin clients to access data and services, execute programs via the Internet and store data in virtual folders. Mobile IT is indeed an important part of the often brought up vision of ubiquitous computing, described as unobtrusive services and applications that are always accessible (Mark Weiser, 1993, 1994).

In the light of this, one could say that IT users have gone from being simply nomadic to being truly mobile, able to access and use the same services constantly, regardless of current place or device. This assumption has many flaws though. Despite the increase in mobile devices used along with new communication technologies (3G, WiMax, new highly improved versions of WiFi etc) and an abundance of web and cloud services, society still lacks the ability to use IT anywhere, anytime. An Internet connection might be missing or broken, trust issues might prevent a user from consuming different services (such as virtual storage) and the metaphor of the cloud might not be intuitive to all users. There are still areas within mobile computing that are not fully explored; areas that might contain concepts that help complement the ambition of systems that fully support mobile IT usage. One such concept could well be Context Awareness Supported Application

Mobility. Taking this as a point of departure, the purpose of this paper is to conceptually advance “application mobility” towards design, i.e. to explore the concept in search for solutions that increase mobility.

1.1 Related Work on Context Awareness Supported Application Mobility

The basic notion of “mobility” has been applied on different aspects of the IT usage context, e.g., terminal mobility, session mobility (mobility of information and media streams), service mobility or personal mobility (ITU, 2002). Applications can also be mobile. In 1995, Bharat and Cardelli presented migratory applications as:

“A new genre of user interface applications that can migrate from one machine to another, taking their user interface and application contexts with them, and continue from where they left off. Such applications are not tied to one user or one machine, and can roam freely over the network, rendering service to a community of users, gathering human input and interacting with people.” (p. 133)

Migratory, or mobile, applications thus resemble mobile agents, but the latter are typically non-interactive and function as silent agents on the user’s or client’s behalf. Mobile applications are not restricted to certain types of programs, and they also always have user interfaces and keep all of their states when migrating (Bharat & Cardelli, 1995). Migration can be either sender initiated, when the user or the system actively sends the application to a new host device, or receiver initiated, when the application is instead fetched onto the new host (Cui, Nahrstedt, & Xu, 2004). Regardless of migration type, the application first has to be paused, then moved (using a wired or, in mobile scenarios, preferably wireless communication technology) and finally made to continue its execution on the new device, thus being completely removed from the original host. Zhou et al. (2007) name these stages suspension, migration and resump-

tion. When transferring an application between different devices during its execution, one has accordingly achieved *application mobility*.

Application mobility always exists within a *context*. Here we use Dey and Abowd's definition, describing context as "*any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves*" (1999, pp. 3-4). Making a system capable of sensing its context and then act according to its own reasoning upon the data describing the situation, is making it *context-aware*. Combining Dey and Abowd's (1999) definitions of context awareness with state of the art definitions of application mobility (e.g., Koponen, Gurtov, & Nikander, 2005), the concept of *Context Awareness Supported Application Mobility* (henceforth referred to through the acronym CASAM) could be described as when using context to move an application between different devices during its execution, to provide relevant information and/or services, where relevancy depends on the user's task.

1.2 Purpose and Disposition

The purpose of this paper is to conceptually advance application mobility towards design. In more specific terms that means to *elaborate* and *examine* the concept of CASAM, *testing the viability* of the concept by defining it, *manifesting* it within a prototype, *exposing* it to real users, and *relating* it to state of the art research within the field of mobile systems, according to the method of *concept-driven design research* (Section 2).

In outlining the paper we now present the concept-oriented method guiding our research (Section 2) followed by Section 3, in which we apply the phases of the concept-driven research method to explore, critique and express the notion of CASAM. Following the method of concept-driven design research we then examine CASAM through an external design critique (Section 4), followed by a discussion

of CASAM as mobile IT support for home care groups (Section 5) before concluding the paper in Section 6.

2. METHOD-CONCEPT-DRIVEN DESIGN RESEARCH

There are different methodologies to choose between when designing IT. User-centered approaches, along with participatory design, activity theory, contextual design, and to some extent ethnographic methods, are all examples of empirically oriented approaches, very common in contemporary IT design. A fundamental element of these empirical approaches is that IT design must be derived from thorough analysis of an existing situation. In making a systematic and complete investigation of users and their context, the IT design will emerge somewhat automatically, as the natural answer to the consideration of all variables of the problem. Empirical approaches have certainly proven successful when designing IT artifacts for existing cases and concrete situations.

However, designing IT can also have an impact on constructing theories. Building upon theories of futuristic use scenarios and reasoning grounded in theory (discussed by Weick, 1989), Stolterman and Wiberg (2010) present a complementary design approach, labeled concept-driven design research.

In *concept-driven design research* the striving for new knowledge is structured in 7 basic phases consisting of: 1) Concept generation, 2) concept exploration, 3) internal concept critique, 4) design of artifacts, 5) external design critique, 6) concept revisited, and finally, 7) concept contextualization.

Through the process of 1) *concept generation* new possible concepts are formulated based on earlier theoretical work in the field. This generation of new concepts is typically a process not possible to prescribe: it may be done by working with associations, metaphors, conflicting or opposing theoretical concepts, theories from other design fields, and historical or other paradigmatic examples. The second

process is called 2) *concept exploration*. In this process the researcher works hands-on with materials, creating models and prototypes and experimenting with unusual materials, forms and content in the exploration of new design spaces. The aim with this process is to explore the design spaces given a formulated guiding concept.

In the third process, called 3) *internal concept critique* the strength of the chosen concept is examined before moving on to a more formalized design. In this phase it is, according to Stolterman and Wiberg (2010), important to relate the design and its underlying concepts to the established theoretical foundation. The success of this phase relies on the identification and establishment of (1) the uniqueness of the chosen core concepts and (2) to what extent the concepts relate to existing theory, and (3) how well these concepts can be clearly expressed in a concrete design.

Having established the conceptual ground the fourth process, 4) *design of artifacts* is about the carving out and expressing of a concrete artifact as a manifested composition that incorporates the concept design as a “whole”. This means that the development of the actual manifestation becomes part of the design process and of the theoretical development. This is where concept-driven design research relates theoretical development with the skilled craft of making artifacts that manifest the full meaning of a theoretical concept (Stolterman & Wiberg, 2010).

The fifth process called 5) *external design critique* is about tests and evaluations. In most approaches, testing includes a question of *user acceptance* (as is the case in usability testing from the perspective of user-centered design). In concept-driven design research, testing means instead that the conceptual design is exposed to a public and critiqued as a composition. It is an evaluation of the idea, the concept and the inherent theoretical principles that the design manifests.

Based on the results from the tests and evaluations, the sixth process called 6) *concept revisited* is about revising and refining the con-

cept that has guided the design. In this sense the evaluation in phase 5 serves as a basis for theory development in this process.

Finally, the last part of this method called 7) *concept contextualization* is about relating and valuing this new concept against the current body of concepts and theory in the field, to position it against similar concepts, and to show how it contributes to previous work.

In our research project, as presented in this paper, we have set out to follow the basic steps of this method with a particular focus on the concept of CASAM – Context Awareness Supported Application Mobility, and how it can be conceptually advanced towards design through a process including concept definitions, concept interpretations, concept implementation in the form of a working prototype, and finally refined and further related to the current body of research within this field.

3. CONCEPT EXPLORATION, CRITIQUE, AND EXPRESSION

The concept of CASAM has been studied by several research groups, often from a strictly technical point of view. Projects like Roam (Chu et al., 2004), Sparkle (Siu et al., 2004), Gaia (Ranganathan, Shankar, & Campbell, 2005), MDAgent (Zhou et al., 2007), SAMProc (Schmidt, Kapitza, & Hauck, 2007) and MSP (Højgaard-Hansen, Nguyen, & Schwefel, 2010) resulted in centralized solutions, often with a middleware taking care of communication and controlling migration, in most cases using centralized code loading. Centralized architectures rely on stable network connections to function satisfactory. These systems become very Internet-dependent, which can show to be a weakness when it comes to supporting mobile technologies in contexts where network quality differs or connections sometimes go down.

The Hydra project (Sato, 2004) was based upon a decentralized solution, which had a multicast setup, RFID supported location awareness and possible separation of input, output and logic on different devices. The papers do not

reveal any occurrences of case studies being conducted, nor have real users evaluated the prototypes. Instead evaluations focus on the measurement of suspension, migration and resumption times in laboratory environments. Also, few comparisons with competing or complementing technologies and paradigms are made. Concept discussions in a wider perspective are left out for the benefit of a more technical focus.

To our knowledge, the most recent architectural proposal for supporting application mobility was presented by Johansson, Åhlund and Åhlund (2011). This architecture contains several novel features building on peer-to-peer technology to handle application identification and context management, taking advantage of the overlay network topology for managing nodes and spreading context data over distributed hash tables. By adding Mobile IP to the architecture, usage is not restricted to local networks, and applications can be migrated to any authorized device connected to the network, regardless of location. Thus, the architecture supports full mobility in a global scope, while still containing a decentralized structure.

All these projects and resulting prototypes have a common factor in that they show that application mobility can be achieved, and that there are several different methods possible when designing for application mobility.

3.1 CASAM versus Cloud-Based Virtualization and Migration Services

This section headline is provocative as it puts CASAM head to head with the cloud paradigm. Still, we find it important to clarify the differences between these two ways of supporting mobility; they both have their specific strengths and weaknesses. CASAM will not replace cloud-based application frameworks, rather co-exist and complement other existing mobile technologies.

“The cloud” is a metaphor for the Internet, and cloud computing is essentially about virtualizing software, platforms, infrastructure and/or

hardware, presenting abstractions to the user. Through a single point of access, the user can execute programs, store and retrieve files and make use of different networks. Rimal, Choi and Lumb (2009) make a good compilation of the pros and cons of cloud computing. The single point of entry creates easy access to services. The clients can be thin, as applications can be run and used through normal web browsers, while in reality being executed on external servers. The same applies to all files and data, which can be put in external repositories and fetched onto a thin client whenever needed. This however creates a dependency of a good and reliable connection; offline usage of cloud services is very limited and sometimes impossible. The user also becomes dependent on one or more service providers. If a service provider can not deliver (flagrant examples of major outages are listed in Rimal et al, 2009), the user’s ability to create workarounds is minimized; as the central idea of cloud computing is about abstractions, system transparency is often narrowed down to a minimum. Another challenge for cloud services is security and trust. Users can not control where data is stored, neither can they monitor the status of firewalls, supposedly secure connections and other important factors of keeping data safe. Furthermore, users can not control who is accessing their data, when and in what purpose. Kotz, Avancha and Baxi (2009) formulate an explicit research question for the larger scientific community to address, when making inquiries about how mobile hardware and software architectures should be designed to help protect user privacy.

CASAM on the other hand is more complex to the single user. Services are not abstracted in the same manner as with cloud-based application frameworks or migration services; we need to install programs, monitor their functionality by ourselves and bring or fetch applications, rather than just makes use of a single point of access for all services. The clients must be thicker; strong enough to process code, fast enough to handle memory and have the ability to store the actual applications locally. On the other hand, the importance of reliable Internet

connections is vastly reduced. CASAM could in fact be carried out within a local area network or, in critical cases, through near field communication. Also, a CASAM user does not have to rely upon service providers as the application is running on the local device. When it comes to security, the application moves with the user (or at least according to the user's will), which means that it will not be accessible through entry points other than the ones that the user chooses, most often the single device currently executing the application. The user can also control firewalls and which networks and communication channels to use in broader extent than if relying on cloud services.

3.2 Manifesting the Concept

Our first concrete manifestation of the expressed concept of CASAM is a middleware providing seamless application mobility, while being context-aware and deployed in a decentralized manner. The middleware is called A2M, specified in Åhlund et al. (2009). In short, the system consists of three major components: the migration manager, the application adapter, and the GUI adapter.

The *migration manager* is the core component, acting as device discoverer and server component (lowering the risk of application loss) through multicasting. A context collector module is responsible for the gathering and storing of context information. In the prototype we draw on user location decided through RFID sensor technology to determine user presence. Also embedded within the migration manager component is a mobile application manager module. It allows the component to control migration between devices through TCP sockets. Using Java Reflection, a migrated application can resume its execution on the new device without the destination host having knowledge of the application prior to the migration. The migration manager has to be installed on all presumptive host devices.

The *application adapter* sees to it that the migrated application adapts to the capabilities and restraints of its new device. It is responsible

for keeping the execution states so that the application can continue to execute where it left off when suspended.

Finally, the *GUI adapter* modifies the graphical user interface of the application, so it will suit the new host device.

3.3 Video Conference Application

The fundamental task of the A2M system is of course to migrate applications between devices. To allow the A2M to carry out its intended task, we built a video conference application using Flash and Java. The video conference application allows two users to communicate through both video and audio in real time. The application also provides a chat window for sending asynchronous text messages.

4. EXTERNAL DESIGN CRITIQUE

To further explore a feasible realization and implementation based on the concept of CASAM, we wanted to identify a potential user group which could offer us insights from an outside perspective, basically to empirically situate our concept exploration in practice. For this we needed a suitable research site with a mobile user group. To qualify, the user group had to be mobile in the sense that we among their tasks could find information to be carried between and/or needed in different contexts of their work. We found such a group in an organization called "the Mobile Team", a mobile home care service group.

During a two-day field study, we examined the typical workflow within the Mobile Team. The field study was designed following the directions of contextual inquiry method (Beyer & Holtzblatt, 1998), emphasizing observation in combination with non-formal interviews and conversation during the work situation. Throughout the study the interviewer must interfere as little as possible with the work of the interviewee, preserving a normal work context. Broad and plentiful data gathering for in-depth post analysis is an essential part of the method.

4.1 The Mobile Team

In the Swedish municipality of Skellefteå, around 700 people are employed within the home care service sector. Just over 20 of these people, all certified nursing assistants (CNAs), form a group called the Mobile Team. The team assists the regular home care service in caring for patients that have just been dismissed from hospital, or who need tending at the last stages of life. The CNAs in the Mobile Team work day and night in shifts. The team is (as stated in team name) mobile; every shift consists of seven simultaneous cars, operating within a 20 km radius with a shared office as base.

Group organization is flat; the employees take turns in shouldering the role as planner, deciding how to organize and divide the work for the upcoming two weeks. The work schedule must be updated every morning, as new patients are added, old patients no longer need care, and reorganization if circumstances change, such as if an employee becomes ill. If the planner cannot fill the schedule, he/she compiles available resources and offers these to all 32 substitute coordinators within the municipal home care service in Skellefteå. These resources can then be booked and used by other home care service areas. When the advertisement of these free resources is finished, the planner heads out in the field, attending to the same work as everybody else in the group, while still remaining in the role as planner. A certain amount of planning and scheduling continues constantly, as certain events can occur with short notice. This scheduling must be made ad hoc, putting high demands on the planner.

The current level of IT support is low. There are two stationary computers in the shared office. The computers are used for emailing and for compiling the text documents, containing available resources. All scheduling is made manually, with pen and paper. Communication between employees is carried out face to face in the office, through notes in a dedicated folder placed on the desk in the common office, or via mobile phone. Logs concerning the care of the specific patients being treated are placed in the

residences of the patients, so that the patients, their relatives and the responsible CNAs can catch up on what kind of care has been given. Often a patient is treated by two or more CNAs during the typical two to three weeks of care by the Mobile Team.

4.2 Typical Workflow within the Mobile Team

Every morning, during approximately two hours, the planner schedules care efforts for the upcoming two weeks. Free time slots in the schedule are identified, summarized in a text document and e-mailed to the 32 substitute coordinators within the municipal home care service. The substitute coordinators can at any time return with requests of booking available resources, and the schedule is then updated with these new assignments. At the same time, the text document containing free time slots must be updated, so that a time is not offered twice, causing double booking. Comparing the schedule with the list of available time slots is experienced as a messy and time consuming task by the CNAs.

A paper copy of the schedule for the day is placed on the desk in the shared office. The planner carries with him/her the original schedule throughout the day, updating the schedule ad hoc if needed. This might cause a disturbance if the planner is busy tending a patient. The planner has no possibility to update the copy when in the field.

At the beginning of a shift, when employees arrive at the shared office, they read the schedule to find out which patients to visit. Sometimes they also read the personal file about the patient, stored within the office. Then a key to the assigned car is received, and the CNAs start their work in the field, visiting the patients in their homes. Addresses might be hard to find, especially if the patient lives in the countryside. Arriving at the patient, the CNA reads the patient's log. Finding this log might also be hard, as patients, their relatives and/or the responsible CNAs does not always remember to put it in the same place every time after reading it. The

Table 1. Identified problems

Set of tasks	Identified problem(s)
Scheduling	<ul style="list-style-type: none"> • The schedule is bound to physical locations. • The copy of the schedule cannot be updated from afar. • Changes in the schedule is made with a pen, disallowing rearrangement without first deleting entries • Hard to match free time slots in the schedule with the list submitted to substitute coordinators. • Possible disturbance of the patient if scheduling ad hoc in the field.
Going to the patient	<ul style="list-style-type: none"> • The patient's address might be hard to find.
The care situation	<ul style="list-style-type: none"> • The patient's log might be hard to find.
Communication with other team members	<ul style="list-style-type: none"> • Information in the diary folder is not sorted by subject. • Older pieces of information in the diary folder might be hard to find.
Debriefing	<ul style="list-style-type: none"> • Time reports are sometimes forgotten. • Sometimes hard to remember carried out work. • Often time consuming.

CNA checks the medication list, medicates the patient and might also do some cleaning and/or cooking, depending on the needs of the patient. Comments about the work are written down in the log by the CNA. As the log is kept within the patients home, these comments can be read by both CNAs, patients and relatives, and thus the log can be used for asynchronous communication between the parties.

During the day, most often the CNA will return to the shared office for lunch and/or coffee breaks. When the shift is over (or during the day if the office is visited earlier), notes can be left in the diary folder so that other members of the Mobile Team can read about special events. As this information is written on paper in a chronological manner, historical notes can be hard to find. Also, every CNA is responsible for recording time spent and type of care given at every patient. Sometimes this can be hard to remember (Table 1).

4.3 Concept Evaluation

To evaluate the CASAM concept, a workshop was organized where scientists, developers and personnel from the Mobile Team would participate. The workshop consisted of three phases: 1) concept presentation and field study summary, 2) test of the A2M system that acted as

a manifestation of the CASAM concept and 3) focus group. Attending the workshop were five CNAs from the Mobile Team, all with different attitudes towards new technology and IT usage (a deliberate choice to capture a wider range of opinions); one of the developers of the A2M system; and two senior computer scientists. The workshop was hosted and chaired by the first author of this paper.

4.3.1 Concept Presentation and Field Study Summary

First of all, the concept of CASAM was presented to the workshop participants. Some participants had some prior experience of the technology, but for most participants the subject was new. The notions of application mobility, context and context awareness was explained and demonstrated. A summary of the field study was also presented. During the presentations, the CNAs could ask questions and make comments. The problems identified during the field study were confirmed by the attending CNAs.

4.3.2 Testing the A2M

To further demonstrate the CASAM concept and give the workshop attendees a chance to experience CASAM hands on, a user test session was

carried out. The A2M system, acting as concept manifestation, was installed on two laptops and used in combination with our own-developed videoconference application. The workshop attendees took turns in communicating with each other via the laptops. After a while, one of the users switched to a new device (a mini PC), at the same time automatically migrating the application so that he or she could continue the videoconference from that device.

4.3.3 Focus Group Study

After being introduced to the concept of CASAM, in line with the concept-driven design research method a focus group was conducted with the purpose of evaluating the concept and informing the design ideas. The focus group was deliberately open ended, with as little steering as possible from the workshop host. Many opinions about the prototype and the concept, as well as possible applications were raised and discussed. Among the most important and/or interesting was that the prototype usage was experienced as intuitive with the location connected to an RFID tag, and that migration was considered seamless despite migration times of between 5 and 9 seconds. There was a belief that a CASAM based work support would save time (no need for login, upstart, manual resuming of applications etc) and simplify the current workflow. Worries about security were shown, but mobile devices could be protected with card reader technology, already adopted within the organization. As for artifact design, it should be easy to understand, but the system should also be able to understand the user. Integrating the design with mobile phones was considered important, and a discussion about GPS navigation support was also had. Regarding modalities, video did not emerge as a prioritized means of communication.

5. TOWARDS DESIGN - CASAM AS IT SUPPORT FOR MOBILE HOME CARE GROUPS

An often adapted approach to implementation of IT support within the health care area is digitalization and centralized storage (Avison & Young, 2007; Thompson & Dean, 2009). This could benefit home care groups such as the Mobile Team in some extent. Digitalizing scheduling, maps, journals, diaries etc and storing them within the cloud would in the ideal case allow the CNAs access to important data in the field. However, this solution does not stem from the CASAM concept, and, as powerful it may be, it also has some major drawbacks. In addition to the need for centralized security solutions and the increase in Internet dependency, the mismatch between these kinds of solutions and the work practices they should support is often put forth as a criticism (Avison & Young, 2007).

Following the method of concept-driven design research (Stolterman & Wiberg, 2010) and leaning on the results of the focus group, we have created a specification of an artifact that manifests our desired theoretical ideas of CASAM as to not only conceptually advance “application mobility” but also as an attempt to move from concept exploration towards design. We call our design A-Maid (Application Mobility Aid) and as a design draft it will be further presented in the following section.

The artifact is comprised of several tiers. The first tier consists of network and hardware components (both mobile and stationary) providing the infrastructure needed for transferring applications. The second tier is the middleware, controlling the migration processes and maintaining a peer-to-peer overlay network, used to manage the devices and the context data. The top tier is the actual application. Figure 1 summarizes.

Regarding hardware, all CNAs will be equipped with mobile devices, in this particular case smartphones with interfaces for 3G, WiFi and short-range communication technologies (NFC and/or Bluetooth). GPS functionality is also offered. The smartphones fill two purposes within the A-Maid system: they serve as communication tools, as well as means of using and carrying the application. The CNA responsible for planning also brings a laptop, giving better input and output modalities compared to the smartphones. All patients also have thin clients stationed in their homes. The most important features of the thin clients are big screens (compared to the smartphones'), small local storage and the possibility to communicate with the smartphones via NFC or WiFi. No Internet connection is needed.

As a basis for communication we make use of the technology within the Peer-to-peer (P2P) paradigm (Buford, Yu, & Lua, 2008). We create an overlay network consisting of the smartphones, laptop, thin clients and stationary PC nodes, which can be distributed geographically throughout the area in which the Mobile

Team operates. Full technical details about the proposed P2P architecture are provided in Johansson, Åhlund and Åhlund (2011). The middleware linking application and infrastructure must handle all important communication between the two tiers. It contains a Migration Manager (MM), providing interfaces for the application when it comes to connectivity, context handling and migration control. The MM is installed on all nodes that form the A-Maid net. It also contains an Application Adapter (AA), which embeds the application and sees to that the application adapts according to the specifications received from the system. Finally, a GUI Adapter (GA) help with changing graphical user interfaces, adapting these to the requirements and modalities supported by the device onto which the application will migrate. Much of this is already implemented within the A2M middleware (Åhlund et al., 2009) used during the workshop test session.

The context-aware mobile application is to be seen as a secretary or assistant to the CNA. It contains a calendar tool, a map tool and a journal editing tool. The calendar tool answers

Figure 1. The a-maid artifact

up to the problems connected to scheduling. By swapping the paper schedule with a digital version, bound to the calendar tool, the schedule is taken from the space of physical places to the space of flows (as defined by Castells, 2000). This makes the schedule available to the CNAs in the field and enables new types of editing, not bound to a physical place or the use of a pencil. The planner can edit the schedule by running the application on the laptop, and then either push the schedule data onto other CNAs mobile devices through tunneling (if connected to the Internet) or synchronizing the data when back at the shared office. The possible disturbance of the patient when rescheduling in the field is kept to a minimum when this can be done without making phone calls back and forth. The map tool makes use of the GPS in the smartphones, but can also be used when offline, letting the CNA navigate against maps downloaded beforehand while at the shared office. The most explicit use of CASAM is shown in the journal editing tool. The tool never leaves the CNA, nor does the data; like a digitalized version of a small word processor, the tool follows him or her everywhere, and the data is stored locally.

From the conceptual exploration as reported in this paper we have identified three major advantages of using CASAM to complement solutions of digitalization and central storage:

- 1) Introducing an application that supports current workflow rather than substituting it with a new data flow where the data is separated from the context makes the IT usage more intuitive for the CNA.
- 2) Security is prioritized as important data follows the user (actually the data follows the application, which in turn follows the user). There is never a need to send or store sensitive data in the cloud.
- 3) A-Maid is non-Internet dependent. All that is needed to carry out the migration of the application is a local network, based on radio technology, NFC or Bluetooth.

While these advantages point at specific advantages of using CASAM as a guiding

concept for this particular design, we simultaneously see how this exemplifies and validates the concept-driven design research method as a fruitful approach to the problem at hand.

6. CONCLUSION

The purpose of this paper was to conceptually advance “application mobility” towards design. In more detail to examine the concept of context awareness supported application mobility (CASAM), testing the viability of the concept by defining it, manifesting it within a prototype, exposing it to real users and relating it to state of the art research and technology within the field of mobile systems. This is in accordance with the method of concept-driven design research.

In this endeavor CASAM was defined through identifying and describing the different parts of the concept, being application mobility, context and context awareness. It was then compared to the concept of cloud computing, pointing out the strengths and drawbacks of this kind of IT usage depending on the chosen technology. CASAM was concretely manifested in a prototype called A2M.

The most important contribution from this work was the exploration of the concept through identification and survey of a real work case, consisting of a field study and a workshop, allowing us to expose our concept to a public and through a focus group inform the subsequent creation of a specification of an artifact that manifests our desired conceptual ideas of CASAM. The artifact, A-maid, is a digital CNA assistant characterized by supporting current workflow, prioritizing security and being non-Internet dependent.

To summarize, this work has shown how a mobile IT concept can be formulated, explored and validated through a concept-driven design research approach; how user participation and concept manifestation can help for evaluating the concept; and finally and for this particular case, how IT artifacts derived and developed from and for the concept of CASAM can be used to support the work of home care service

groups. Our focus has been on recognizing and making use of a mobility concept, rather than being about using mobile technology to overcome problems connected to mobile contexts. Therefore, we see our results operating at both a practical level as well as on the level of the conceptual.

ACKNOWLEDGMENTS

This article has been compiled as part of the Mobile and Open Service Access project (MOSA, <http://www.mosaproject.org/>) and the Nordic Interaction and Mobility Research Platform (NIMO, <http://www.nimoproject.org/>), which are both supported by EU structural funds. The authors would like to thank the CNAs of the Mobile Team for participating. The authors would also like to thank Emma Karlsson for her very useful contribution during the field study, and Karl Andersson for much appreciated help and assistance during research and project planning. Our thanks also go to Sarah Fahleson for valuable comments on our manuscript.

REFERENCES

- Åhlund, A., Mitra, K., Johansson, D., Åhlund, C., & Zaslavsky, A. (2009, September 2-4). Context-aware application mobility support in pervasive computing environments. In *Proceedings of the 6th International Conference on Mobile Technology, Application & Systems*, Nice, France (pp. 1-4).
- Avison, D., & Young, T. (2007). Time to rethink health care and ICT? *Communications of the ACM*, 50(6), 69-74. doi:10.1145/1247001.1247008
- Beale, R. (2009). What does mobile mean? *International Journal of Mobile Human Computer Interaction*, 1(3), 1-8. doi:10.4018/jmhci.2009070101
- Beyer, H., & Holtzblatt, K. (1998). *Contextual design: Defining customer-centered systems*. San Francisco, CA: Morgan Kaufmann.
- Bharat, K. A., & Cardelli, L. (1995, November 15-17). Migratory applications. In *Proceedings of the 8th Annual ACM Symposium on User Interface and Software Technology* (pp. 132-142).
- Buford, J., Yu, H., & Lua, E. (2008). *P2P networking and applications*. San Francisco, CA: Morgan Kaufmann.
- Castells, M. (2000). The information age: Economy, society and culture: Vol. 1. *The rise of the network society* (2nd ed.). Malden, MA: Blackwell.
- Chu, H., Song, H., Wong, C., Kurakake, S., & Katagiri, M. (2004). Roam, a seamless application framework. *Journal of Systems and Software*, 69(3), 209-226. doi:10.1016/S0164-1212(03)00052-9
- Cui, Y., Nahrstedt, K., & Xu, D. (2004). Seamless userlevel handoff in ubiquitous multimedia service delivery. *Multimedia Tools and Applications*, 22(2), 137-170. doi:10.1023/B:MTAP.0000011932.28891.a0
- Dey, A. K., & Abowd, G. D. (1999). *Towards a better understanding of context and context awareness* (Tech. Rep. No. GIT-GVU-99-22). Atlanta, GA: College of Computing, Georgia Institute of Technology.
- Hojgaard-Hansen, K., Nguyen, H. C., & Schwefel, H. (2011). Session mobility solution for client-based application migration scenarios. In *Proceedings of the Eighth International Conference on Wireless On-Demand Network Systems and Services* (pp. 76-83).
- ITU. (2002). *ITU-T Recommendation H.510, mobility and collaboration procedures. Mobility for HSeries multimedia systems and services*. Retrieved from http://www.itu.int/rec/dologin_pub.asp?lang=e&id=TREC-H.510-200203-1!!PDF-E&type=items
- ITU. (2011). *ICT statistics database*. Retrieved from <http://www.itu.int/ITU-D/ict/statistics>
- Johansson, D., Åhlund, A., & Åhlund, C. (2011). A MIP-P2P based architecture for application mobility. In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia*, Beijing, China (pp. 85-93).
- Kaikkonen, A. (2009). Mobile Internet: Past, present, and the future. *International Journal of Mobile Human Computer Interaction*, 1(3), 29-45. doi:10.4018/jmhci.2009070104
- Kleinrock, L. (1996). Nomadcity: Anytime, anywhere in a disconnected world. *Mobile Networks and Applications*, 1(4), 351-357.
- Koblentz, E. (2009). How it started: Mobile Internet devices of the previous millennium. *International Journal of Mobile Human Computer Interaction*, 1(4), 1-3. doi:10.4018/jmhci.2009062601

- Koponen, T., Gurtov, A., & Nikander, P. (2005, May). Application mobility with HIP. In *Proceedings of the 24th International Conference on Thermoelectrics*.
- Kotz, D., Avancha, S., & Baxi, A. (2009, November 13). A privacy framework for mobile health and home-care systems. In *Proceedings of the First ACM Workshop on Security and Privacy in Medical and Home-Care Systems*, Chicago, IL (pp. 1-12).
- Ranganathan, A., Shankar, C., & Campbell, R. (2005). Application polymorphism for autonomic ubiquitous computing. *Multiagent and Grid Systems*, 1(2), 109-129.
- Rimal, B. P., Choi, E., & Lumb, I. (2009, August 25-27). A taxonomy and survey of cloud computing systems. In *Proceedings of the Fifth International Joint Conference on INC, IMS and IDC* (pp. 44-51).
- Satoh, I. (2004). Self-deployment of distributed applications. In N. Guelfi, G. Reggio, & A. Romanovsky (Eds.), *Proceedings of the Fourth International Workshop on Scientific Engineering of Distributed Java Applications* (LNCS 3409, pp. 48-57).
- Schmidt, H., Kapitza, R., & Hauck, F. J. (2007, March 20). Mobile process-based ubiquitous computing platform: A blueprint. In *Proceedings of the 1st Workshop on Middleware-Application interaction held in conjunction with Euro-Sys*, Lisbon, Portugal (pp. 25-30).
- Siu, P. P. L., Belaramani, N., Wang, C.-L., & Lau, F. C. M. (2004, August 26-28). Context-aware state management for ubiquitous applications. In *Proceedings of the International Conference on Embedded and Ubiquitous Computing*, Aizu, Japan (pp. 776-785).
- Stolterman, E., & Wiberg, M. (2010). Concept-driven interaction design research. *Human-Computer Interaction*, 25(2), 95-118. doi:10.1080/07370020903586696
- Thompson, S. M., & Dean, M. D. (2009). Advancing information technology in health care. *Communications of the ACM*, 52(6), 118-121. doi:10.1145/1516046.1516077
- Weick, K. E. (1989). Theory construction as disciplined imagination. *Academy of Management Review*, 14(4), 516-513.
- Weiser, M. (1993). Hot topics: Ubiquitous computing. *IEEE Computer*, 26(10), 71-72. doi:10.1109/2.237456
- Weiser, M. (1994). The world is not a desktop. *Interaction*, 1(1), 7-8. doi:10.1145/174800.174801
- Zhou, Y., Cao, J., Raychoudhury, V., Siebert, J., & Lu, J. (2007, June 22-29). A middleware support for agent-based application mobility in pervasive environments. In *Proceedings of the 27th International Conference on Distributed Computing Systems Workshops* (p. 9).

Dan Johansson is a PhD student in Mobile Systems at the Department of Computer Science, Electrical and Space Engineering at Luleå University of Technology, Sweden. His research interests include mobile systems, application mobility, context awareness, mobile services and applications.

Mikael Wiberg, PhD is a Chaired Professor in Human-Computer Interaction at the Department of Informatics and Media at Uppsala University, Sweden. His research interests include mobile and ubiquitous systems, Mobile IT for group collaboration and novel approaches to interaction design.