To appear in Proceedings of fifteenth International Conference on Distributed Multimedia Systems (DMS'09), Redwood City, San Francisco Bay, USA, 2009/9

LoCa – Towards a Context-aware Infrastructure for eHealth Applications*

Nadine Fröhlich[1] Andreas Meier[2] Thorsten Möller[1]
Marco Savini[2] Heiko Schuldt[1] Joël Vogt[2]

[1] Department of Computer Science, University of Basel, Switzerland
[2] Department of Informatics, University of Fribourg, Switzerland

E-mail: {nadine.froehlich, thorsten.moeller, heiko.schuldt}@unibas.ch
{andreas.meier, marco.savini, joel.vogt}@unifr.ch

Abstract

New sensor technologies, powerful mobile devices and wearable computers in conjunction with wireless communication standards have opened new possibilities in providing customized software solutions for medical professionals and patients. Today, medical professionals are usually equipped with much more powerful hardware and software than some years before. The same is true for patients which, by making use of smart sensors and mobile devices for gathering, processing and analyzing data, can live independently in their home environment while receiving the degree of monitoring they would get in stationary care. All these environments are highly dynamic, due to the inherent mobility of users. Therefore, it is of utmost importance to automatically adapt the underlying IT environment to the current needs of their users – which might change over time when user context evolves. In a digital home environment, this requires the automatic customization of user interfaces and the context-aware adaptation of monitoring workflows for mobile patients. This paper introduces the LoCa project which will provide a generic software infrastructure, able to dynamically adapt user interfaces and services-based distributed applications (workflows) to the actual context of a user (physician, caregiver, patient). In this paper, we focus on the application of LoCa to monitoring the health state of mobile patients in a digital home environment.

1. Introduction

Telemonitoring applications enable healthcare institutions to control therapies of patients in out-of-hospital settings. In particular, telemonitoring allows patients to live as independently as possible in their digital home environment. The goal is to support the individual disease management by patient monitoring which will result in less hospitalization and a higher quality of life. In the presence of an increasingly aging population and a growing number of people suffering from chronic ailments, this kind of applications already has a high relevance for the healthcare system and is expected to gain even more importance.

Monitoring includes the continuous gathering, processing and analysis of mainly physiological data coming from sensors which are either integrated into the patient's digital home or attached to the patient's body or clothes. Currently, these monitoring applications are rarely automated. Configuration of the sensor environment, the customization for a particular patient, and the actual data processing and analysis are mostly tedious manual tasks. In the LoCa project (A Location and Context-aware eHealth Infrastructure), we aim at providing a user-friendly and adaptable solution for the automated gathering and analysis of relevant data for monitoring patients. LoCa will be a general purpose system that can be applied both in digital home environments and in stationary care. A main feature in LoCa is the consideration of context as a first class citizen. This means that monitoring applications and processes as well as user interfaces will be dynamically adapted based on the user's context (e.g., location, activity, etc.). Context-aware adaptations will result in more customized monitoring solutions and thus better support for data analysis and emergency assistance (e.g., triggering of emergency services in case of severe health conditions). Dynamic adaptations will also allow to seamlessly apply best practices in health monitoring and patient control without explicit reconfigurations.

Consider, for instance, a sixty-five year old male patient with cardiac problems in convalescence. During his recovery at home, his physician would like to control his state of health and therefore needs to continuously receive data on his physiological condition. At the moment, the patient's ECG is measured periodically once a day or additionally, in

*The LoCa project is funded by the Hasler Foundation.

case the patient does not feel well. For this, a nurse is sent to the patient's home to record ECG data and other measurements. The physician only receives raw data and has to manually initiate all the steps needed for the interpretation of raw data in a particular order, including a comparison of the actual values with the patient's medical history, to determine the individual development of physiological data.

In order to improve this situation, the patient is given a smart shirt equipped with several sensors metering physiological parameters like ECG and blood glucose level. In addition, the patient receives a smart phone with GPS sensor and camera. From the point of view of the patient, this allows for almost unlimited mobility and does no longer require him to stay at home for the necessary measurements. From the physician's point of view, the smart shirt allows for the continuous gathering of vital parameters and thus for seamless monitoring in real time. As an important requirement for properly analyzing and interpreting metered data, the physician needs to know the exact context of the measurement (e.g., the patient's location and activity). Therefore, the shirt not only has to provide physiological data but also details on his activity (e.g., by means of acceleration sensors that can monitor the physical exercises he is doing). The patient's therapy includes a healthy diet, without alcohol and cigarettes, as well as physical exercises he is not used to. Thus, he writes an electronic diary, extended with photos of his meals, which finally helps in communicating diet information and stress factors to his physician. Annotations to this diary, provided by the physician, support the patient in understanding effects of his behavior for his therapy. Having access to raw sensor data does not yet allow the physician to properly analyze the patient's health state. The data still has to be cleaned, eventually coarsened, and analyzed in correlation with each other. For data analysis, the physician will follow a process consisting or dedicated processing steps in pre-defined order. To ease her work she will use the LoCa system to define these workflows in a user-friendly way, thereby determining rules for data interpretation. Finally, she is able to define proper thresholds, for instance for critical blood pressure values in stress situations.In case a threshold is exceeded, the physician will be visually advised on her screen or will receive an SMS. It is important to note that neither the analysis processes nor the corresponding user interfaces are static but need to be automatically adapted as soon as the context of the patient changes (e.g., when a different set of sensors is available), or in the course of the therapy when further parameters need to be taken into account.

The objective of the LoCa project is to address the challenges introduced above and provide reliable support for workflow-based eHealth applications. This includes telemonitoring in home care as well as applications in stationary care. In close collaboration with stakeholders from the healthcare domain, different use cases from both applications have already been defined. Finally, the LoCa system will be applied and evaluated in a stationary care and in a home care environment by the medical project partners. In this paper, we focus on telemonitoring applications in a digital home environment. From a functional perspective, the goal is to gather, process, analyze, and visualize physiological data and to store aggregated data in the electronic health record of a patient. In particular, the analysis and visualization will be dynamically tailored to the patient's context. This includes sophisticated failure handling which, by considering context at run-time, does not need to be prespecified in monitoring workflows. The system should finally be able to detect and anticipate potential cardiac irregularities or other health-related problems, based on criteria defined by the medical partners in the project. From a systems point of view, LoCa will make use and extend an existing platform for the reliable processing of data streams for health monitoring across fixed and mobile devices [5, 6].

In this paper, we present the ongoing LoCa approach to context-aware monitoring applications in digital homes. An important constraint in this scenario is that users (patients) are mobile, which means their context might frequently change. Therefore, the way data — coming from different soft- or hardware sensors — is analyzed needs to be automatically adapted, if necessary. The same is true for the interaction of the user with the system. The basis of these adaptations is a powerful context model and its exploitation to dynamically adapt i.) user interfaces and services and ii.) process-based distributed applications (workflows).

The remainder of this paper is organized as follows: Section 2 introduces the LoCa context model. The architecture of the LoCa system is presented in Section 3. In Section 4, we discuss context-aware adaptation in LoCa. The status of the current implementation is presented in Section 5. Section 6 surveys related work and Section 7 concludes.

2. Context Model

LoCa exploits a generic context model to improve health care applications and to facilitate the treatment of patients, both in home care and in stationary care. To reach this goal, we need to adapt processes and user interfaces automatically according to the current context. This, in turn, necessitates the proper representation of context information. We have designed a generic context model for context data management. Figure 1 depicts this model in Entity-Relationship notation. In here, we closely follow the well established definition of context by Day et al. [1]: *Context is any information that can be used to characterize the situation of a subject. A subject is a person, place, or object that is considered relevant to the interaction between a user and an application [...].*

To appear in Proceedings of fifteenth International Conference on Distributed Multimedia Systems (DMS'09), Redwood City, San Francisco Bay, USA, 2009/9

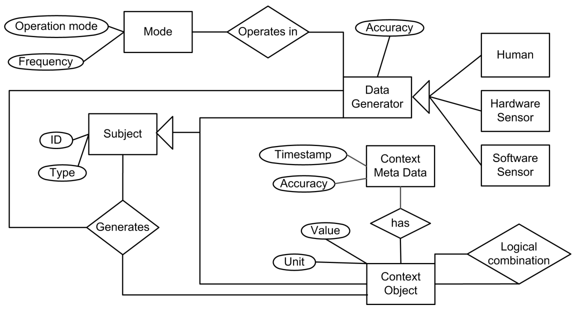

Figure 1. LoCa Context Model

The *Subject* can be a patient, a mobile phone, or an ECG sensor. Conversely, profile data, the medical history, current ECG data, or the current location are examples for context information about a patient. The entity *Context Object* represents the actual context data, e.g., the value of the current location, a document of the medical history, and so on. In order to support data analysis, we store optional meta data about context objects, such as time stamps and data accuracy (which usually depends on the type of sensor used).

The entity *Data Generator* (humans, hardware sensors, software sensors) is designed to capture data about the instrument (sensor) which produces context data: a data generator generates context data about subjects. While many data generators generate atomic data, some sensors may produce compound context objects. For instance, the (GPS) location usually consists of multiple values, such as longitude, latitude, altitude, speed, and bearing. Furthermore, software sensors can combine different kinds of context objects to compose higher level context data. An alarm in case of cardiac problems could be combined of information about the current activity of a patient and his current ECG values. This is covered in the model by means of the relationship *logical combination*.

The context model is able to handle different kinds of context objects, including nested context objects. An important feature of the context model is its rather simple, yet expressive structure. It is powerful enough to cover all the different context objects that have been identified in the requirements analysis phase of LoCa in which several home care and stationary care use cases have been analyzed together with stakeholders from the eHealth domain. Nevertheless, the model can be extended by adding new data generators and thus also new context objects, if necessary.

3. Architecture of the LoCa Platform

Context awareness requires that the information gathered from distributed sensors is stored in a global, albeit distributed database on the basis of the schema presented in Sec. 2. Prior to inserting raw sensor data into this database, it needs to be cleaned and transformed into the global schema. Since context data is a vital input for all LoCa applications, the context data management layer forms the basis of the LoCa architecture depicted in Figure 2.

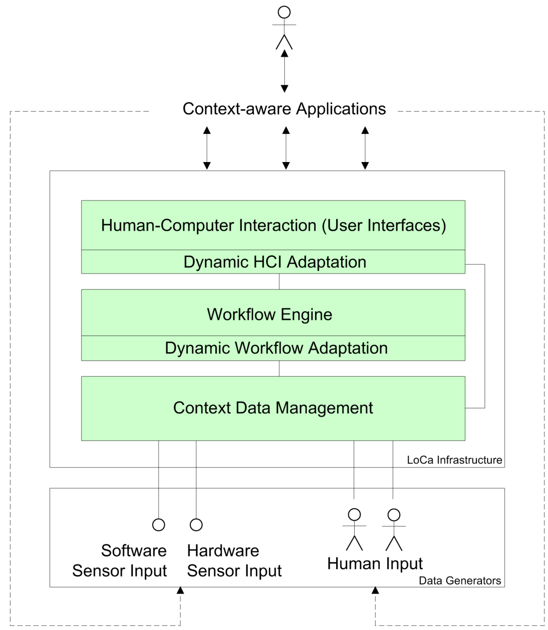

Figure 2. LoCa Conceptual Architecture

On top of context management, the LoCa applications are defined as workflows. The basic assumption is that functionality is available in the form of (web) services so that workflows can be defined by combining existing services. Since complete workflows again have a service interface, service composition can be applied recursively. A crucial part of this layer is dynamic workflow adaptation. This layer makes use of the raw sensor data and their relationships stored in the context layer. The top-most layer of the LoCa architecture deals with the dynamic generation and adaptation of user interfaces. Again, this layer directly accesses the underlying context data management.

All layers are embedded in the LoCa infrastructure which is described in more detail in Section 5. The LoCa architecture offers a unified interface for (individual, user-defined or pre-existing) workflow based applications. According to the context model, LoCa workflow-based applications themselves can be considered software sensors, i.e., they might produce context objects which are subsequently needed for dynamic adaptation.

4. Context-aware Adaptation in LoCa

In what follows, we address the dynamic adaptation needed in LoCa for applications in the eHealth domain, namely at workflow (process) and at user interface level.

4.1. Context-aware Workflows

Traditional approaches to workflow management usually consider static settings as they can be found in business processes or office automation. However, these approaches are far too rigid to handle highly dynamic environments as they can occur in the medical domain, especially when monitoring mobile patients in their (digital) home environment. From a workflow management perspective, these applications are characterized by a potentially large number of i.) exceptions or unforeseen events (e.g., abnormal deviations in sensed physiological data that may require alternative medication); ii.) different ways to achieve a goal (e.g., different devices can be used to meter blood pressure); iii.) decisions only decidable at run-time (e.g., results of tests cause different subsequent tests or treatments); and iv.) dynamic and continuous changes (e.g., new devices, or treatment methods).

Context-aware, adaptable workflows offer much more flexibility than traditional workflows as they allow for structural changes based on evolving user context. Basically, structural changes of workflows can be done at build-time (prior to the instantiation of workflow processes) and at run-time (changing an instance of a workflow). Build-time changes cover evolutionary changes of processes but also changes caused by context changes like new methods of treatments, hospital guidelines, laws, etc. These kinds of workflow changes are not in the primary focus of LoCa. We will mainly address run-time changes such as, for instance, allergic hypersensitivity of patients that cause changes in the treatment process (e.g., adding an allergy test).

There are two kinds of run-time changes [19] — process adaptation and built-in flexibility. Process adaptation, that can be performed at run-time, is based on modification operations like add, delete, or swap of process fragments. Built-in flexibility supports the exchange of process fragments of a workflow. For instance, assume the examination of a special disease differs depending on the age of the patient because the risk to get this disease and its severity increases with the age of the patient. Thus, the examination always follows the same basic structure while the concrete steps depend on the patient's risk group. Therefore, a workflow consisting of placeholders and concrete steps is defined at build-time. Steps that differ depending on the age are defined as placeholder activities and steps that not differ as usual activities. At run-time, placeholder activities are replaced by the concrete fragments depending on the patient's risk group.

Variants of built-in flexibility are described in [19]. Three of them are of particular importance for the eHealth applications in LoCa: i.) late selection, ii.) late modeling, and iii.) late composition. They differ in the degree of decision deferral and need for user experience. The least flexibility is offered by late selection where workflows, defined at build-time, contain placeholder activities that are substituted by a concrete implementation during run-time. Late modeling additionally supports modeling of placeholder activities at run-time. The most flexible pattern is late composition. At build-time, only process fragments are specified. At run-time workflows are composed out of the process fragments available. In LoCa, we will adopt late composition and will make use of the services' semantics (using semantic web service standards) for the actual selection.

Applied to the scenario presented in Section 1, the treatment workflow has to be adapted dependent on the vital parameters of the patient. Assume that the therapy is less successful than expected so that the physician decides to also meter the blood pressure of the patient. In this case the workflow for controlling the patient's health state has to be extended accordingly. Usually, the physician is informed about irregularities in the patient's ECG values by visually highlighted values and, if severe problems occur, by an SMS to his mobile phone. The extension to a new sensor requires also the adaptation of the signal processing and triggering.

In LoCa, we focus on run-time changes of workflows without manual intervention. Particularly, we will provide rules for automated adaptation of workflows, that is, automated fragment selection or composition based on user context and service semantics.

4.2. Context-aware User Interfaces

Adapting user interfaces in a context-aware environment allows the various actors of the system the best possible utilization of the available resources. Therefore simply defining one standard user interface (UI) design and adapting it to the display of the device the user is currently using will not be sufficient [10, 13, 21].

In LoCa, each user interface component (i.e., button, pulldown menu, picture) will be described in an artifact and be interpreted at run-time. This generic description contains the type of the component, its position within a hierarchy, a mapping to the environment that allows listening to incoming information and a label.

Another artifact with a set of rules is responsible for mapping the generic composite to a concrete representation for a given situation. This rendering mechanism is executed at run-time in order to choose the currently most optimal way to display the component. It takes into account the following contextual information: i.) *device:* information about the current device, such as displaying capabilities, current network bandwidth and latency, CPU usage, remaining battery time, etc. This might be a mobile device of the patient or any device of the patient's digital home environment; ii.) *user:* who is using the current device. This information may also cover several users, such as the doctor and a patient during a ward visit; iii.) *location:* the current location of the device may also influence the rendering of a component; iv.) *reason:* the reason why a component is displayed may be difficult to obtain. Possible elements of such an information may be the current calendar entries or tasks, the current patient situation, such as ECG; v.) *time:* the dimension time is not simply a timestamp, but may also include time spans or semantical information, such as "after lunch" or "night".

For the application scenario presented in Section 1, this means for instance that the patient's mobile device knows, by making use of the calendar stored on it, that a specific process needs to be started. The device displays the input fields required to enter the required physiological parameters. If the input field for the blood oxygen saturation value is able to find a viable hardware sensor in its proximity (oximeter), it automatically reads the value from that device, sets itself immutable and moves to the bottom of the display. The mandatory input fields that cannot be processed automatically must be filled in by the patient. Each input component must also decide how to react if, for example, the patient fills in a value before it could find a matching hardware sensor in its environment.

5. Implementation

The implementation of the LoCa infrastructure is currently ongoing. LoCa will use and further advance the open service-oriented infrastructure OSIRIS[1] NEXT (ON)[2]. Originally based on the hyperdatabase vision [16], many ideas from process management, peer-to-peer networks, database technology, and Grid infrastructures were integrated in the past in order to support distributed and decentralized process management [18]. More recent work aims at i.) support for distributed data stream management [5, 6] and ii.) the integration of semantic technologies to enable new ways for flexible and automated process management support. This includes support for distributed and decentralized execution of processes in dynamic (mobile) environments [11] as well as an advanced method to enable automated forward-oriented failure handling [12].

In the context of the LoCa project we will exploit and extend the process management system that has been integrated into ON. It allows for dynamically distributed and decentralized execution of composite semantic services that are described based on OWL-S. On top of this, the user interface will be built based on the Android platform[3].

ON essentially represents a P2P-based open service infrastructure. At its bottom layer it realizes a message-oriented middleware enabling arbitrary services which are deployed at peers to interact by means message exchange. Besides the possibility for end-to-end interactions, the platform also realizes a publish-subscribe messaging paradigm. Furthermore, it incorporates advanced concepts for eager and lazy data replication, taking into account user specified data freshness properties. The platform provides several built-in system services that are used to manage meta and runtime information about the services offered by the peers in the network [18].

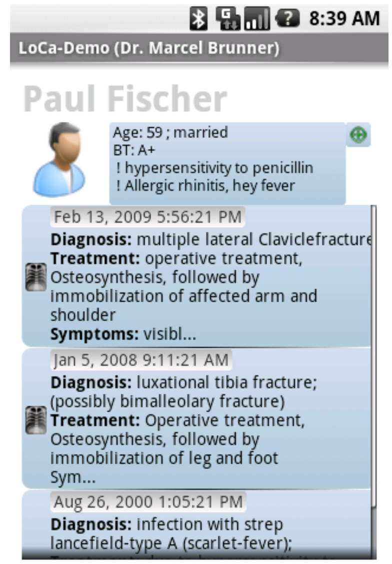

Figure 3. Screenshot of LoCa Demonstrator

ON is fully implemented in Java. One of its key properties is its a small systems footprint (in particular regarding memory) and its internal design is strictly multithreaded in order to take advantage of multi-core CPU technology. Every service spawns its own thread group. Internal message processing is similar to the SEDA approach [20]. It can be deployed in a stand-alone mode on a wide range of devices, starting from mobile platforms, netbooks, up to enterprise computing machines. Moreover, ON can also be deployed as an agent in the JADE[4] agent platform, thus, enabling FIPA compliant usage.

For evaluation and demonstration of our approach, especially of our use cases, we are building a prototype based on Android cell phones. Figure 3 shows an early prototype of the user interface for a physician.

6. Related Work

In the last years, a number of projects have been carried out in the eHealth domain. In particular, many projects

[1] Open Service Infrastructure for Reliable & Integrated process Support
[2] http://on.cs.unibas.ch
[3] http://source.android.com
[4] http://jade.tilab.com

apply workflow and process technology for distributed application in eHealth. Akogrimo [8] deals with the support of dynamic virtual organizations that require the ability to change its structure dynamically and to access data from mobile resources. ADEPT [15] allows to dynamically change the type of workflow instances in order to react to changes in the application (e.g., patient's therapy). While ADEPT addresses mainly change patterns, CAWE (Context Aware Workflow System) [3] deals with built-in flexibility.

A number of eHealth projects also take into account context. The MARC project [2] provides a passive monitoring system that can be used for elderly people. CodeBlue [7] explores various wireless applications in the eHealth domain with a focus on 3D location tracking. ARCS [17] addresses user interface adaptation in eHealth applications. It provides web-based interfaces mainly for stationary devices for manual disease monitoring. In [4], eHealth applications and services to support mobile devices have been designed.

Online monitoring and streaming data is more and more emerging in eHealth. The MyHeart project [14] monitors cardio-vascular parameters using measuring wearable devices (i.e., devices that are integrated into clothes). The PHM project [9] measures different vital parameters either continuously or at determined time intervals.

7. Conclusion and Future Work

LoCa is an ongoing effort that will provide a novel approach to context and location-aware eHealth applications as they can be found when monitoring physiological data and activity status of patients in a digital home environment. By providing generic support for the context-aware adaptation of workflows and user interfaces, LoCa is intended to be applied to other scenarios as well, e.g, in stationary care.

In close collaboration with healthcare practitioners and experts from industry, we have identified several concrete scenarios. The requirements coming from there will be considered when completing the implementation of the LoCa system based on the ON platform. Finally, these scenarios will be evaluated together with our medical partners.

References

[1] G. Abowd, A. Dey, P. Brown, et al. Towards a Better Understanding of Context and Context-Awareness. In *Proc. Int'l Symp. on Handheld and Ubiquitous Computing*, pages 304–307, London, UK, 1999. Springer.

[2] M. Alwan, S. Kell, B. Turner, et al. Psychosocial Impact of Passive Health Status Monitoring on Informal Caregivers and Older Adults Living in Independent Senior Housing. In *Proc. ICTTA'06*, pages 808–813, 2006.

[3] L. Ardissono, R. Furnari, A. Goy, et al. A Framework for the Management of Context-aware Workflow Systems. In *Proc. WEBIST 2007*, pages 80–87, 2007.

[4] J. Bardram. Applications of Context-Aware Computing in Hospital Work – Examples and Design Principles. In *Proc. ACM SAC*, pages 1574 – 1579, 2004.

[5] G. Brettlecker and H. Schuldt. The OSIRIS-SE (Stream-Enabled) Infrastructure for Reliable Data Stream Management on Mobile Devices. In *Proc. SIGMOD*, pages 1097–1099, June 2007.

[6] G. Brettlecker, H. Schuldt, and H.-J. Schek. Efficient and Coordinated Checkpointing for Reliable Distributed Data Stream Management. In *Proc. ADBIS'06*, pages 296–312, Thessaloniki, Greece, 2006.

[7] T. Gao, D. Greenspan, M. Welsh, et al. Vital Signs Monitoring and Patient Tracking Over a Wireless Network. In *Proc. IEEE EMBS*, pages 102–105, 2005.

[8] T. Kirkham, D. Mac Randal, J. Gallop, and B. Ritchie. Akogrimo — a Work in Progress on the Delivery of a Next Generation Grid. In *Proc. SOAS'05*, 2005.

[9] C. Kunze, W. Stork, and K. Müller-Glaser. Tele-Monitoring as a Medical Application of Ubiquitous Computing. In *Proc. MoCoMed'03*, pages 115–120, 2003.

[10] P. Langley. User Modeling in Adaptive Interfaces. In *Proc. UM'99*, pages 357–370. Springer, 1999.

[11] T. Möller and H. Schuldt. A Platform to Support Decentralized and Dynamically Distributed P2P Composite OWL-S Service Execution. In *Proc. MW4SOC'07*. ACM, 2007.

[12] T. Möller and H. Schuldt. Control Flow Intervention for Semantic Failure Handling during Composite Serice Execution. In *Proc. ICWS'08*, pages 834–835, 2008.

[13] E. G. Nilsson, J. Floch, S. O. Hallsteinsen, and E. Stav. Model-based user interface adaptation. *Computers & Graphics*, 30(5):692–701, 2006.

[14] M. Pacelli, G. Loriga, N. Taccini, and R. Paradiso. Sensing Fabrics for Monitoring Physiological and Biomechanical Variables: E-textile solutions. In *Proc. Int'l Symposium on Medical Devices and Biosensors*, pages 1–4, 2006.

[15] M. Reichert, S. Rinderle, and P. Dadam. On the Common Support of Workflow Type and Instance Changes Under Correctness Constraints. In *Proc. CoopIS'03*, pages 407–425, Catania, Italy, Nov. 2003. Springer LNCS.

[16] H.-J. Schek and H. Schuldt. The Hyperdatabase Project - From the Vision to Realizations. In *Proc. BNCOD*, pages 207–226, Cardiff, UK, 2008.

[17] G. Schreier, A. Kollmann, M. Kramer, et al. *Computers Helping People with Special Needs*, volume 3118, pages 29–36. Springer, 2004.

[18] C. Schuler, C. Türker, H.-J. Schek, R. Weber, and H. Schuldt. Scalable Peer-to-Peer Process Management. *Int. J. of Business Process Integration & Management*, 2006.

[19] B. Weber, S. Rinderle, and M. Reichert. Change Patterns and Change Support Features in Process-Aware Information Systems. In *Proc. CAiSE 2007*, pages 574–588. Springer LNCS, Trondheim, Norway, June 2007.

[20] M. Welsh, D. Culler, and E. Brewer. SEDA: An Architecture for Well-Conditioned, Scalable Internet Services. In *18th Symp. on OS Principles*, Banff, Canada, 2001.

[21] Z. Yu, X. Zhou, D. Zhang, et al. Supporting Context-Aware Media Recommendations for Smart Phones. *IEEE Pervasive Computing*, 5(3):68–75, 2006.