Models in the Design of Context-Aware Well-Being Applications

Steven Bosems and Marten van Sinderen

University of Twente, Department of EECMS, Enschede, The Netherlands
{s.bosems,m.j.vansinderen}@utwente.nl

Abstract. Context-aware systems that make use of sensor information to reason about their context have been proposed in many domains. However, it is still hard to design *effective* context-aware applications, due to the absence of suitable domain theories that consider dynamic context and associated user requirements as a precursor of system development. In this paper, we discuss a theory for the *well-being* domain and propose a model-driven development process that exploits the proposed theory to build effective, i.e. user-centric, context-aware applications.

Keywords: Well-Being, Context-Aware, Model-Driven, Causal Reasoning.

1 Introduction

We define context-aware computing as the combination of sensor, reasoning and other technology that provides IT systems with real-time awareness of their environment and advanced analytics to offer services that address the varying requirements of their users (cf. [2]). Widely used alternative terms to describe this paradigm are smart, ambient, ubiquitous, and pervasive computing [6, 9].

Despite many technological advances in this area, the design of effective context-aware applications is hard because designers cannot always anticipate the dynamics of context and associated user requirements [8].

In this paper, we propose a theory to address this problem, albeit limited to the well-being domain. Furthermore, we present a model-driven development process that applies the proposed theory to establish user requirements for selected well-being goals and derive context-aware well-being applications to support these goals.

This paper builds on the research presented in [3]. The paper is further structured as follows: Section 2 introduces our theory of well-being; Section 3 presents a model-driven development process for context-aware well-being applications using the theory; Section 4 explains how validation of developed applications may lead to improvements to both the theory and these applications; Section 5 concludes the paper, mentioning related work, our contributions so far, and planned future research.

R. Meersman et al. (Eds.): OTM 2014 Workshops, LNCS 8842, pp. 37–42, 2014.

2 Theory of Well-Being

Our theory of well-being is intended to be used for the development of context-aware applications that support individual well-being. This is in contrast to most existing theories that try to answer philosophical questions about what is 'good for' a person [7, 15]. Our theory is based on the assumption that a person's well-being is influenced by interactions between health conditions and contextual factors (cf. [17]). Certain conditions and factors can be monitored by IT applications through sensors, and reasoning algorithms based on knowledge on the interactions can be used to decide on useful interventions (informing, advising, instructing a person, or controlling certain conditions or factors). We will only consider those conditions, factors and interactions that are proven by medical practice, which classifies our theory as objective [15].

Our theory is expressed as a causal loop diagram (a special directed graph) consisting of nodes and edges. Nodes represent variables for health conditions and contextual factors, and edges represent cause-effect relations between two variables. Nodes may have boundary values for the variables, including min and max values, and objective norms. Edges have a polarity to indicate whether a change of the 'source' variable causes a change in the same ('+') or opposite ('-') direction for the 'target' variable. Other attributes may be used to characterize the timing between cause and effect (e.g., immediate or delayed) and the strength of the effect (e.g., weak or strong).

We built our theory by studying literature and interviewing experts. This resulted in the identification of a set of relevant variables and their causal relations, all mapped into a single causal loop diagram, which we call the Dynamic Well-being Domain Model (DWDM) [5]. We realize that the knowledge that we have captured is still very limited. Therefore, we consider it as a 'living' model, which is subject to constant improvement based on our own new insights and other people's comments.

The modeling constructs used for the DWDM are simple but powerful. They allow to reason about the importance and effect of conditions and factors for well-being goals, where goals are statements that can be related to one or more of the variables. For example, a goal can be to improve one's physical condition, and a factor that has a (delayed) positive effect on this is physical exercise. One interesting phenomenon that is exposed by the model is that of feedback loops. A reinforcing loop exists if the loop contains an even number of causal relations with '-' polarity. These loops reinforce themselves in that a change in any variable will lead to a self-fueled increasing change in all variables in the loop (provided all remains the same outside the loop). A balancing loop exists if the loop contains an odd number of causal relations with '-' polarity. These loops are self-correcting as a change in any variable propagates and returns a change of that variable in the opposite direction. Both types of loops are illustrated in Fig. 1, showing an excerpt of our DWDM.

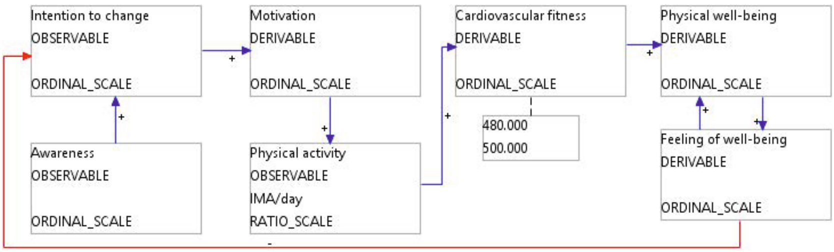

Fig. 1. Excerpt of DWDM relevant for improving well-being through physical exercise

3 System Development

The DWDM is system-independent, but it can be used to constrain and steer the development of context-aware applications [3]. We distinguish four development phases: (i) selection of relevant variables, (ii) design of a first system model, (iii) design of a refined system model, and (iv) deriving the application code. These phases follow the Model Driven Architecture (MDA) in the sense that they correspond with the CIM, PIM, PSM and code level, respectively, of MDA [11].

Selection of Relevant Variables. In order to develop an application that supports some well-being goal, we have to identify the variables (and relations) in the DWDM that are relevant to the selected well-being goal [14]. For example, Fig. 1 shows the variables that are relevant to the goal of improving physical well-being through physical exercise. This could be used to develop an application that monitors and gives advice on physical activity (in fact, the model was inspired by an existing application, called Activity Coach [13]). The model can be specialized by considering the characteristics and preferences of a target user group (e.g., athletes, elderly, or workers) or even an individual user.

Design of First System Model. Subsequently, we analyze the model from the previous phase to determine which variables can be directly measured through (affordable) sensors or through user interaction. For example, in Fig. 1, “Physical activity” can be measured by an accelerometer/gyroscope sensor and “Awareness” can be measured by querying the user. We call such variables observable. There are also variables that cannot be directly measured. To quantify these variables, we have to use the values of variables related to them. We call such variables derivable. These classifications are of course not fixed, but depend on available budgets and the technological state of the art.

Further, we have to decide in what way the well-being goal will be supported, keeping in mind that components exist that correspond to observable variables. For example, the goal of improving physical well-being can be supported by providing monitoring information and advice on physical activity. In this case, the provided information is based on the interpretation of data from the accelerometer/gyroscope, and the advice is based on the variation in observable variables and what effect the propagation of this variation causes in other variables more directly related to the goal.

We adopt a general context-aware system architecture, such as the one described in [4], and use the decisions mentioned above to instantiate this architecture, resulting in a first system model. The model may comprise several views. For example, a structural view can show the accelerator/gyroscope and a user dialog component as well as other components that follow from a mapping of the expected support onto the generic structure of the architecture. Next to that, a behavioral view can show the interactions between or involving these components, partly based on the need for communication to realize the expected support to the well-being goal.

Design of Refined System Model. The next step is to decide how the components will be distributed and which technology platforms will be involved. Since the system handles personal and private information, the user must trust the system and all entities involved in the systems operation. To mitigate a potential distrust, the number of involved service providers can be limited. We identify three types of platforms/services: (i) smartphones, (ii) external storage providers, and (iii) web/application servers. A user might be more trusting if information is kept on the smartphone, rather than offloading storage and/or computation to external parties. On the other hand, limited power and storage resources on the smartphone have to be taken into account.

Once a strategy of distribution has been chosen, platform specific bindings can be defined. This entails adding information about technology choices and infrastructure services to run the previously identified components.

Deriving Application Code. The refined model can be translated to application code, the language of which depends on the platform(s) of choice. Code generation for specific platforms has become common practice, so we will not further discuss this step here.

4 Theory/System Co-evolution

Throughout the development process of a context-aware application, decisions are constrained or informed by the DWDM. Assuming that both our DWDM and the design decisions are correct, the application will provide the expected support to the chosen well-being goal. If either is incorrect, the application will likely not provide useful behavior for the well-being goal. If the latter is the case, the application and the system models need to evaluated for adherence to the DWDM. If a discrepancy is found, the application has to be modified. If no discrepancy is found, the DWDM is probably based on incomplete or faulty knowledge, and has to be modified. Next, the system development has to be repeated with the improved DWDM.

The means and content of the system-provided interventions to the user (in our example, information and advice) is crucial for user acceptance, and therefore also for the experienced support to the well-being goal. Since the DWDM cannot be used for these aspects of the system design, we need here involvement of behavioral scientist and HCI developers, the discussion of which is outside the scope of this paper.

5 Conclusions and Future Work

Context-aware systems have a clear potential to provide better user services in many domains. But it is hard to realize this potential. Often, developed applications are not effective in that they fail to support user goals in certain (unforeseen) contexts encountered by the user. Establishing and understanding user requirements in the design phase, and systematically developing an application from these requirements while applying design patterns and platform knowledge are recognized problems in this area [1, 6, 9, 10, 12, 16].

In this paper, we have introduced a theory of well-being and a Dynamic Well-being Domain Model (DWDM) to help understand user requirements for well-being goals and to constrain and steer the development of applications to support such goals. The proposed development process is an application of the Model Driven Architecture (i.e., model-driven development). The DWDM captures knowledge on cause-effect relations between health conditions and contextual factors, and makes it possible to reason how changes in these variables can contribute to a well-being goal. Understanding the nature of the variables and the effect of their variation at the user level can be translated to design decisions in the different phases of the development process.

We continuously try to improve our DWDM by considering new studies and experiences as well as input from experts. We are also investigating the possibility of automating parts of the development process. The proposed approach is currently evaluated through experiments, in which two groups of students (one using tools that support our process, the other a control group) design and document a context-aware application.

Acknowledgements. This publication was supported by the Dutch national program COMMIT (project P7 SWELL).

References

[1] Alférez, G.H., Pelechano, V.: Dynamic evolution of context-aware systems with models at runtime. In: France, R.B., Kazmeier, J., Breu, R., Atkinson, C. (eds.) MODELS 2012. LNCS, vol. 7590, pp. 70–86. Springer, Heidelberg (2012)
[2] Bartels, A.H.: Smart computing drives the new era of it growth. Forrester Inc. (2009)
[3] Bosems, S., van Sinderen, M.J.: Improving context-aware applications for the well-being domain: Model-driven design guided by medical knowledge. In: 3rd Int. Conf. on Sensor Networks (SENSORNETS 2014), pp. 397–403 (2014)

[4] Bosems, S. et al.: SWELL D1.2 Overall architecture. Tech. rep. (2013)

[5] Bosems, S. et al.: Dynamic domain model for well-weing (2014), http://wwwhome.ewi.utwente.nl/~bosemss/2014/OTMA/ddm.png

[6] Dey, A.K., Abowd, G.D.: The context toolkit: Aiding the development of context-aware applications. In: SIGCHI Conf. on Human Factors in Computing Systems, pp. 431–441 (1999)

[7] Easterlin, R.A.: Building a better theory of well-being. In: Economics & Happiness: Framing the Analysis, pp. 29–64 (2006)

[8] Greenberg, S.: Context as a dynamic construct (2001)

[9] Henricksen, K., Indulska, J.: Developing context-aware pervasive computing applications: Models and approach. Pervasive and Mobile Computing 2(1), 37–64 (2006)

[10] Kailas, A., et al.: From mobile phones to personal wellness dashboards. IEEE pulse 1(1), 57–63 (2010)

[11] Kleppe, A.G., et al.: MDA explained: The model driven architecture: Practice and promise. Addison-Wesley (2003)

[12] Lane, N.D., et al.: BeWell: A smartphone application to monitor, model and promote wellbeing. In: 5th Int. ICST Conf. on Pervasive Computing Technologies for Healthcare, pp. 23–26 (2011)

[13] op den Akker, H., et al.: Development and evaluation of a sensor-based system for remote monitoring and treatment of chronic diseases - the continuous care & coaching platform. In: 6th Int. Symposium on Ehealth Services and Technologies (2012)

[14] Soriano Perez, C.S.: Context aware systems in well-being: From user needs to software architecture. Master's thesis, Universitat Politecnica de Valencia (2014)

[15] Varelius, J.: Objective explanations of individual well-being. Journal of Happiness Studies 5, 73–91 (2004)

[16] Welsh, K., et al.: Towards requirements aware systems: Run-time resolution of design-time assumptions. In: 26th IEEE/ACM Int. Conf. on Automated Software Engineering, pp. 560–563 (2011)

[17] World Health Organization: Towards a common language for functioning, disability and health: ICF. International Classification 1149, 1–22 (2002)