Human Interaction and Emerging Technologies (IHIET 2022), Vol. 68, 2022, 365–373
https://doi.org/10.54941/ahfe1002754

A Model-Based Environment for Building and Running Situation-Aware Interactive Applications

Christian Märtin[1], Jürgen Engel[1], and Christian Herdin[2]

[1]Faculty of Computer Science, Augsburg University of Applied Sciences, 86161 Augsburg, Germany
[2]Department of Computer Science, University of Rostock, 18051 Rostock, Germany

ABSTRACT

In this paper we present SitAdapt, a development- and runtime-environment for the construction of situation-aware interactive applications that can be automatically adapted in real-time. The system architecture includes a fine-grained user monitoring system that is continually recording emotional, eye-tracking and bio-physical user data as well as application meta data. Adaptations occur when significant situational changes trigger adaptation rules that lead to modifications of the user interface or the behavior of the application. The operation of the system that is co-operating with the PaMGIS MBUID-framework and the steps of the adaptation process that also exploits HCI-patterns are demonstrated for a travel-booking web application. The paper also briefly discusses ethical and privacy issues associated with user monitoring.

Keywords: Situation-awareness, Adaptive systems, Model-based user interface development systems (MBUID), Task model, HCI-patterns

INTRODUCTION AND RELATED WORK

Situation-aware software systems open unique perspectives for improved goal achievement characteristics for both users and software suppliers. Such interactive systems give users individualized user experience, provide runtime-adaptive user interfaces, allow for situation-dependent task planning capabilities, and can trigger service activations when the situational context changes.

Situation-awareness is an evolution of the concept of context-aware computing that was first proposed for distributed mobile computing in (Schilit and Theimer, 1994). In addition to software responsiveness when dynamically migrating applications to other devices and locations, the notion of context then also included aspects related to the interacting people and the environment.

The term situation-awareness was first established in psychology and the cognitive sciences. It was used for describing support systems for human operators in complex situations. By defining situation-dependent requirements the correct and smooth interactive task accomplishment can be facilitated (Flach, 1994), (Flach et al. 2004).

For such systems, considering the changes of the interactive context over time plays an important role. The complete perception process of the users depends on contextual information encountered during the interactive processing of tasks.

To capture the individual requirements of a situation, Chang in (Chang, 2016) has suggested that a situation specification must cover the user's operational environment *E*, the user's social behavior *B*, by interpreting his or her actions, and a hidden context *M* that includes the user's mental states and emotions. A situation *Sit* at a given time *t* can thus be defined as $Sit = <M, B, E>_t$. A user's intention for using a specific software service for reaching a goal can then be formulated as a temporal sequence $<Sit_1, Sit_2, \ldots, Sit_n>$, where Sit_1 is the situation that triggers the usage of a service or a task execution and Sit_n is the goal-satisfying situation.

This model provides a good starting point for the construction of situation-aware applications. There are two different, but strongly related views on situation-awareness. Situation-awareness as discussed in (Mulder et al. 2017) and many other approaches, supports the user to correctly interpret her current situation by letting her focus on the most important contextual and situation-typical information, e.g., in complex and safety-critical human-machine interactions. The other view, as taken in our approach, is a situation-analytic view, where the system is observing the user, the interactive software, and the environmental and platform-related context. The system recognizes specific situations and combinations of situations and adapts the system in real-time to optimize task accomplishment and user experience.

However, for recognizing and evaluating situations, the hidden context *M*, i.e., the user's emotions, intentions and cognitive load must be made visible and exploitable. Adaptive situation-aware software systems need cognitive and analytic capabilities to interpret the multitude of available emotional and bio-physical signals as well as application and contextual meta-data to being able to infer a goal-reaching set of adaptations.

Rather than providing situation-awareness separately for each developed application, we have designed an integrated model-based environment for building and running dynamically adaptive situation-aware applications. Therefore, we follow a combined approach by linking the *PaMGIS (Pattern-Based Modeling and Generation of Interactive Systems)* MBUID-framework (Engel et al. 2015) and its domain and user interface models with *SitAdapt*, a high-resolution observation platform with an integrated user-centric adaptation component (Märtin et al. 2019). SitAdapt offers functionality for visual emotion recognition (Noldus FaceReader), eye- and gaze-tracking (Tobii eye-tracker), pulse-rate-tracking (Empatica wrist band), EEG-signal recording (g.tec brain-computer interface), as well as various other bio-physical signal monitoring capabilities. Situations, i.e., the varying observed values of the user's emotional, eye-tracking, bio-physical and interactive behavior during task execution are recorded with a minimal resolution of 1/60s and stored in situation profiles in a database. Situation profiles are currently recorded in a lab environment where test subjects interact with the applications and are monitored before, during and after the sessions. Information about a user that can be gathered automatically before a session or provided interactively by the user is stored in a user model.

For integrating the necessary reasoning capabilities, we have introduced situation rules in SitAdapt that have access to the situation profiles, the user profiles, and the modeling resources of the MBUID-framework. Situation profiles and user profiles can be evaluated in the LHS of the situation rules and trigger actions or runtime-modifications of the user interface and dialog behavior of the interactive application.

The modeling resources are used for constructing the original user interface and the modifications triggered at runtime by situation rules. They include models that define the structure and functionality of, e.g., web applications and include task and domain models of the application, HCI-patterns as well as design and layout models for generating the interactive parts of the source code.

In the following sections we will demonstrate the different steps of the building process and the adaptation process that creates a priori adaptations based on the parsing of user and environment models and dynamic runtime-adaptations that are triggered by situation rules fired when a user-specific situation profile encounters conditions that were specified in the situation-rule model of the application.

We have tested the SitAdapt system with different e-commerce web-applications, e.g., an existing beauty products shopping portal that was enhanced with situation-awareness capabilities and runtime adaptivity (Märtin et al. 2021). In this environment we also conducted two user studies for evaluating the monitoring features and adaptive power of the system with respect to functionality and user experience.

In this paper we will present and discuss the construction process and runtime-operation of an adaptive situation-aware web-application. Examples from a travel-booking application are used to demonstrate how HCI-patterns and individual adaptations are used to tailor the application to the needs of a specific user. We will conclude the paper with a discussion of ethical and privacy issues arising when confronting users with such powerful monitoring systems. We will also give a short outlook on assessing situation-rule performance and optimizing the rules by implementing a reinforcement learning-component in the future.

THE PAMGIS FRAMEWORK

The PaMGIS framework uses the domain- and context models as proposed by the CAMELEON reference framework (CRF) (Calvary et al. 2022). This reference framework also includes structural guidelines for adapting software to the requirements of different platforms and for migrating an application from a device to another device. The current version of the PaMGIS framework architecture is based on (Engel et al. 2015). By co-operating with SitAdapt, it can react to context changes in the user's environment as well as situational changes that include the user's behavior and emotional state.

The PaMGIS Models

The PaMGIS framework uses different models (see Figure 3) for the generation of user interfaces. The *abstract user interface model (AUI)* is generated by using information from the task model based on *Concur Task Trees (CTT)* (Paternò, 2000) and the concept model contained in the domain model. The AUI includes the specifications of the abstract user interface objects. In the domain model and the originally rendered AUI the user interface is still independent of the usage context. After the completion of AUI modeling, the AUI model can be transformed into a *concrete user interface model (CUI)* in the next step. The information of the context model and the structure of the dialog model are exploited by this process. For defining the dynamic aspects of the user interface, PaMGIS uses a dialog model. The dialog model is based on *dialog graphs* that were originally introduced by the TADEUS system (Elwert and Schlungbaum, 1995). In the next step the *final user interface model (FUI)* is generated from the CUI model. In this step information from the design/layout model, the toolkit model and the context model is integrated into the generation process.

To simplify the creation of the different models in the PaMGIS framework, HCI-patterns, pattern fragments and UI-templates based on these patterns are available as in the PaMGIS pattern and model repositories. In the following sub-sections, we demonstrate the construction process for some parts of a travel-booking web application.

The "Display Utilization" Pattern

As a decision criterion for the selection of a certain travel connection, the utilization of the respective means of transport shall be displayed to the passenger. For this purpose, the "display utilization" pattern is used, which is formally described using the *PaMGIS Pattern Specification Language (PPSL)*. A pattern basically consists of the three elements, i.e., the description of the problem to be solved by the pattern, the solution to this problem, and the information about the context in which the pattern can be applied.

The problem of the "display utilization" pattern is described as follows: When booking travel, especially in times of a pandemic, the capacity utilization of the respective transport system can be an important decision criterion when booking tickets. How does the booker get the relevant information?

The solution to this is specified as: The information about the utilization of the means of transport is transmitted and presented together with the other booking data. Since this is often only an estimate, depending on the desired transportation system, the occupancy rate is displayed in categories, e.g., low, medium, or high. These categories can also be conveniently communicated in the form of graphics.

And the application context states: The pattern can be used for booking of travel tickets for passenger transportation by the usual means of transport, such as train, bus or aircraft.

Examples of the use of the pattern are illustrated in Figure 1. On the left side, there is an extract from the booking portal of Deutsche Bahn

Figure 1: Examples for the application of the "display utilization".

```
<Deployment>
 <PaMGIS>
  <ModelFragments>
   <ModelFragment>
    <MDFR_Type>Concept</MDFR_Type>
    ...
    <MDFR_Fragment>
     <Concept>
      <CCPT_ConceptID>"__CPT_0008_01_0001"</CCPT_ConceptID>
      <CCPT_ConceptName>"Transportation Occupancy"</CCPT_ConceptName>
      ...
      <CCPT_ConceptType>Graphics</CCPT_ConceptType>
      ...
     </Concept>
    </MDFR_Fragment>
   </ModelFragment>
  </ModelFragments>
 </PaMGIS>
</Deployment>
```

Figure 2: Excerpt of the PPSL specification of the concept model fragment of the "display utilization" pattern.

AG (Deutsche Bahn, 2022), on the right from the one of Flixmobility AG (FLiXBUS, 2022).

In PaMGIS, patterns can be equipped with model fragments which serve as building blocks for the design of task, concept, or dialog models (Engel et al. 2015). The "Display Utilization" pattern comes with a concept model fragment, which includes a concept that allows to display a graphic indicating the degree of utilization of the respective means of transport. An excerpt of the PPSL specification of this concept model fragment is shown in Figure 2.

When the pattern is selected and applied in PaMGIS, the concept is automatically inserted into the concept model of the application to be implemented. The pattern will be used in the adaptation process that is discussed below.

The SitAdapt System

The SitAdapt system that was linked with the PaMGIS framework offers a situation analysis and rule model for development and runtime support for multiple-adaptive user interfaces and can react to context changes in the user's environment, the user's behavior, and emotional state in real-time. SitAdapt uses two different data types: atomic data types as constant attribute values (e.g., *age range* = 30-35) and temporal data types (e.g., *blood pressure*). With the aid of the SitAdapt rule editor, such atomic and temporal variables can be used to create situation-rules that influence the adaptation of the user interface.

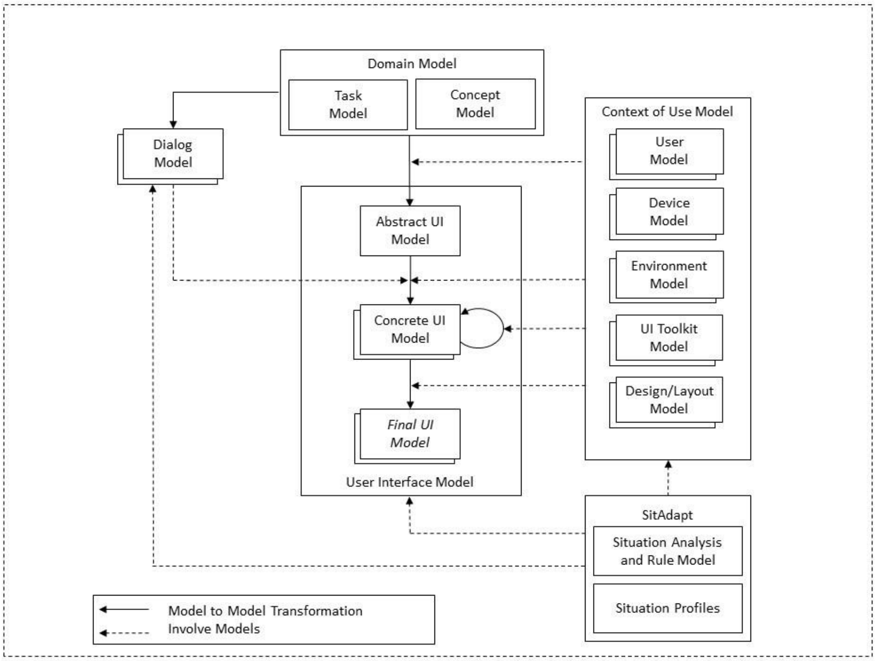

Figure 3: Interaction between PaMGIS modeling components and SitAdapt modeling.

For an adaption of the user interface, SitAdapt requires the following components:

- The *situation analytics component* analyzes and assesses situations by exploiting the observed data that have been stored as situation profiles in the SitAdapt database.
- The *rule editor* allows the definition and modification of rules, e.g., for specifying the different user states and the resulting actions. The rules are triggered by the situation analytics component when the rule conditions are satisfied.
- The *evaluation and decision component* uses the data that are provided by the situation analytics component to decide, whether an adaptation of the user interface is currently meaningful and necessary. Whether an adaptation is meaningful depends on the predefined purpose of the situation-aware target application. If the decision component decides that a complex adaptation is necessary, it must provide the artifacts from the PaMGIS pattern and model repositories and access the context of use models to allow for the modification of the target application by the adaptation component.

The SitAdapt Adaptation Process

The different types of adaptations (e.g., changes of media and interaction object types, content, design and layout, structure) have different effects on the models in the PaMGIS framework. The models must be updated or re-generated.

To demonstrate the possibilities of the adaptive PaMGIS framework, an experimental travel portal based on different existing travel booking portals was developed. During user sessions the current user activity and the completion-state of the travel-booking process can always be queried by mapping the observed user interactions and application meta data to the subtasks of the CTT task model of the travel portal. The SitAdapt travel platform offers various options to adapt to individual users. To make individual adaptations, SitAdapt must first collect information about the users.

For example, during the Covid-19 pandemic, it can be useful for risk groups, such as elderly people, to get additional information for the planned travel. With the help of the API of the FaceReader software the *situation analytics component*, e.g., receives data about the age range of the user and stores it in the user model. In the rule editor domain-specific adaptation rules for users over 50 years could exist.

In this example, the user (over 50 years) is in the early part of the booking process and fills the first form with the basic travel information, such as place of departure and arrival, and the travel date. During this subtask, the FaceReader software observes the user and determines the possible age range of the user in the background and stores this information in the user model. In the next subtask the user wants to display the search results on the screen. The adaptation takes place before the first display of the new content. The evaluation and decision component uses the data that are provided by the situation analytics component and the applicable rules for the modification of the search results page.

For the generation of the search results, the *evaluation and decision component* chooses the special dialog model and design/layout model for users over 50 years. In this dialog it is possible to display additional information for travel connections. In this example the "display utilization" pattern is attached to this subtask. From the dialog model and the AUI with the attached pattern the CUI can be created in the next step. In the following step the FUI is generated from the CUI model by consulting the UI-toolkit and design/layout model. The design model defines the color scheme and the type of diagram presented. The newly generated user interface contains additional information about the utilization of each individual connection (see figure 4). In addition, a graphical diagram is given to the user with the respective occupancy rates of the individual means of transport on this day to provide a possible decision-making aid for a means of transport.

CONCLUSION

In this paper we have discussed some relevant aspects of the construction and operation of situation-aware web applications that were developed within the PaMGIS MBUID-framework and monitored and runtime-adapted by the SitAdapt system.

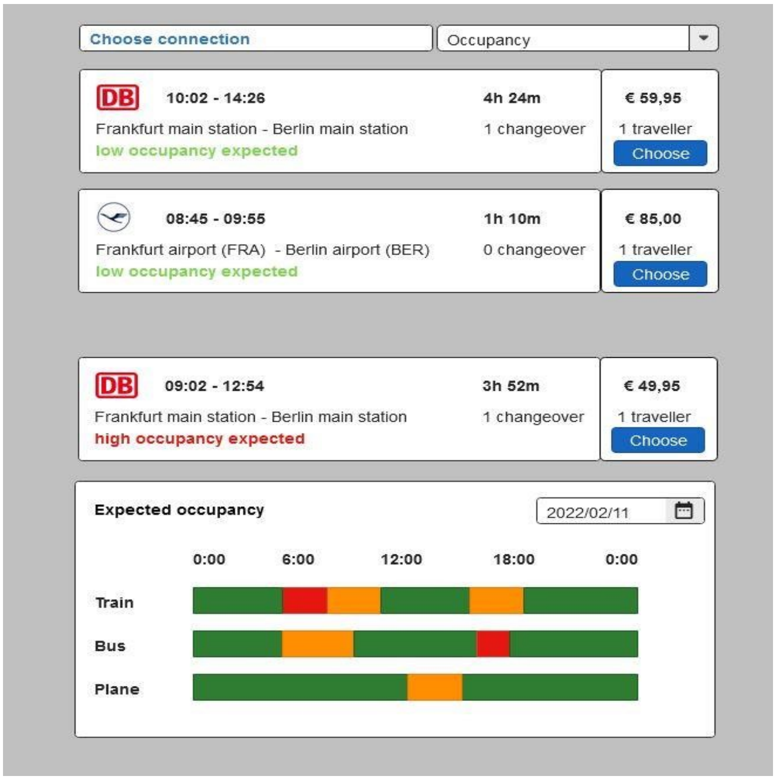

Figure 4: FUI of the search results for age range 50–60.

In our prototyping environment user tests are still carried out under strictly supervised lab conditions. However, currently a rapid evolution is underway of visual and sensor-based emotion recognition technology and its hard- and software integration into all device types from desktop computers to notebooks, tablets, smartphones, and other personal devices. The benefits generated from individualized adaptations that are aware of user emotions and personal characteristics will be experienced by the users only if the recorded data can be securely kept privately, and generic rules of value-sensitive design (Friedman and Hendry, 2019) are followed by the developers and providers of such critical software services. We will direct part of our future research efforts into this direction to allow for better user experience and task accomplishment without compromising the users' privacy.

Another direction for future research we would like to address is the integration of a machine learning component into the SitAdapt component. Given the characteristics and structure of the interactive systems under observation, where a user encounters various situations while striving to accomplish a sequence of subtasks to reach a final goal, e.g., complete her overall travel-booking task, the quality of the situation rule base, i.e., the composition of the adaptation rules, may play a major role. We are currently evaluating, whether Extended Classifier System (XCS), a powerful form of reinforcement learning, could be used as the basis for implementing such a learning component (Stein, 2017).

REFERENCES

Calvary, G., Coutaz, J., Bouillon, L. et al. (2002) "The CAMELEON Reference Framework", http://giove.isti.cnr.it/projects/cameleon/pdf/CAMELEON%20D1.1RefFramework.pdf

Chang, C.K. (2016) Situation Analytics: A Foundation for a New Software Engineering Paradigm, IEEE Computer. pp. 24–33.

Deutsche Bahn (2022), Travel Portal, Website: https://www.bahn.de

Elwert, T., Schlungbaum, E. (1995). Modelling and Generation of Graphical User Interfaces in the TADEUS Approach. In: Palanque, P., Bastide, R. (eds) Design, Specification and Verification of Interactive Systems '95. Eurographics. Springer, Vienna. https://doi.org/10.1007/978-3-7091-9437-9_12

Engel, J., Märtin, C., Forbrig, P. (2015) A concerted model-driven and pattern-based framework for developing user interfaces of interactive ubiquitous applications. In: Proceedings of the First International Workshop on Large scale and Model-Based Interactive Systems, Duisburg, pp. 35–41.

F. Paternò (2000) "The ConcurTaskTrees Notation," in Model-Based Design and Evaluation of Interactive Applications, pp. 39 – 66, Springer Berlin Heidelberg

Flach, J.M. (2003) Situation Awareness: The emperor's new clothes, Mouloua, M., Parasuaman, R. (Eds.) Human performance in automated systems: Current research and trends, Erlbaum, pp. 241–248.

Flach, J.M., Mulder, M., van Paassen, M.M. (2004) The Concept of the Situation in Psychology, in: Branbury, S. et Tremblay, S. (Eds.): A Cognitive Appraoch to Situation Awareness: Theory and Applications, Ashgate Publishing, Oxon (UK), pp. 42–60.

FLiXBUS (2022), Booking Portal, Website: https://www.flixbus.de

Friedman, B., Hendry, D.G. (2019) Value Sensitive Design: Shaping Technology with Moral Imagination, MIT Press, Cambridge Massachusetts

Märtin, C., Bissinger, B.C. and Asta, P. (2021) Optimizing the digital customer journey—Improving user experience by exploiting emotions, personas and situations for individualized user interface adaptations, Journal of Consumer Behaviour, Jun. 2021, doi: 10.1002/cb.1964

Märtin, C., Kampfer, F., Herdin, C., Biawan Yameni, L. (2019) Situation Analytics and Model-based User Interface Development: A Synergetic Approach for Building Runtime-Adaptive Business Applications, Complex Systems Informatics and Modeling Quarterly (CSIMQ) eISSN: 2255-9922 Published online by RTU Press, https://csimq-journals.rtu.lv Article 115, Issue 20, September/October 2019, Pages 1–19 https://doi.org/10.7250/csimq. 2019–20.01

Mulder, M, Borst, C., van Paassen, M.M. (2017) Designing for Situation Awareness – Aviation Perspective, Proc. of the International Conference on Computer-Human Interaction Research and Applications (CHIRA 2017), SCITEPRESS, pp. 9–21.

Schilit, B.N., Theimer, M.M. (1994) Disseminating Active Map Information to Mobile Hosts, IEEE Network, vol. 8, no. 5, pp. 22–32.

Stein, A. (2017) Reaction Learning, in: Müller-Schloer, C., Tomforde, S.: Organic Computing – Technical Systems for Survival in the Real World, Springer International Publishing GmbH, pp. 287–327.