Patterns for HMI design of multi-modal, real-time, proactive systems

Nádia Ferreira
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
nfe@namahn.com

Sabine Geldof
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
sg@namahn.com

Tom Stevens
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
ts@namahn.com

Tom Tourwé
Sirris – ICT & Software
Engineering group
A. Reyerslaan 80, B-1030
Brussels, Belgium
Tom.Tourwe@sirris.be

Elena Tsiporkova
Sirris – ICT & Software
Engineering group
A. Reyerslaan 80, B-1030
Brussels, Belgium
Elena.Tsiporkova@sirris.be

ABSTRACT

The paper reports on the findings of our quest for existing multi-modal design patterns. Further, it describes our approach to elicit new ones from a diversity of real-world applications and our work on organizing them into a meaningful pattern repository using a set of pre-defined parameters, so that they can be described in a uniform and unambiguous way easing their identification, comprehensibility and applicability. These patterns enable designers to optimize the interaction between human operators and systems that reason about and proactively react on information captured e.g. via sensors. Therefore we think that research on interaction with smart objects could benefit of this work.

Author Keywords

Human-centred Interface Design; Interface Design Methodology; Multi-modal Design Patterns; Adaptive Interfaces; Intelligent Support for Complex Tasks; Pro-active Paradigms for User Interaction.

ACM Classification Keywords

H.5.2: User interfaces, H.1.2 User/Machine Systems, D.2.2 Design Tools and Techniques- User Interfaces.

General Terms

Human Factors; Design.

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA Copyright is held by the author/owner(s)

INTRODUCTION

Rationale

The design of multi-modal, adaptive and pro-active interfaces for complex real-time applications requires a specific approach in order to guarantee that the interaction between human and computer remains natural. In order for the interface to adapt to the user and the context, the system needs to reason about her needs and proactively adapt to these while keeping the user in control. The HMI (human-machine interface) design should accommodate varying forms of interaction, depending on what is most appropriate for that particular user at that particular time. HMI design patterns are a powerful means of documenting design know-how, so that it can be re-used. We propose a formal framework to organize and annotate this know-how so that the designer (or at runtime, the system) is supported in the selection (and instantiation) of a pattern, fit to the situation at hand.

Project context

The contribution described in this paper has been developed in the context of ASTUTE[1] a large EU research project. The project focuses on the design of intelligent user interfaces, providing pro-active decision support to the user. The ultimate goal is to develop a platform for building embedded products that capture, reason and act upon user intentions thereby taking into account the user's context (i.e. user environment and all the factors which will influence the user performance) and state (i.e. aspects determining the ability of the user to perform in a given

[1] ASTUTE Pro-active decision support for data-intensive environments; http://astute-project.eu/

situation, such as stress level, fatigue, …). The project approach will be validated with various demonstrators proposed by leading industrial partners in the following domains: automotive (infotainment decision support system aiming at increased driver's safety and comfort), avionics (anticipation support for the pilots to plan their tasks for the descent, under a huge workload), emergency dispatching (distributed and highly dynamic decision support for fire-fighters, their commander and a crisis team facilitating the management of emergency events), building management (context-tailored support for the daily activities of the staff of a rest home) and manufacturing process management (distributed control room solution supporting the different roles on the production floor via distributed devices, both fixed and mobile). Our goal in this project is to provide an appropriate methodology for user interface design, based on design patterns. For this purpose, we have developed a generic approach for collecting and organizing HMI design patterns, so as to facilitate the design of human-centred intelligent interfaces.

Human-centred interaction design and design patterns

In recent years, the need for human-centred design in the development of embedded systems has been recognised ([1], [2], [3] [4]). In need of a systematic approach to their activities, human-machine interaction (HMI) designers are developing human-centred design (HCD) methodologies ([5] [6], [7]). Throughout these methodologies HMI design patterns play an important role in exploring and specifying the interaction between the human user and the computer- in that they inspire design and enable to reuse concrete solutions through appropriate descriptions [8]. The idea of a pattern as a piece of proven and reusable design knowledge has already been applied to interaction design by a number of organizations, resulting in pattern libraries [9], [10], [11], [12], [13], [14] and a key reference publication [15]. More specifically, design patterns for multi-modal interaction have also been investigated in [16], [17], [18], [19]. That collection, far from complete, triggered us to find new patterns and to develop a method to refine their description.

Section 2 describes how we have been collecting patterns, both from the literature and from the demonstrator designs. Section 3 explains our methodology work on organizing the collection of patterns. Section 4 provides an outlook to future activities to continue this research.

COLLECTING PATTERNS

In this section, we describe how a collection of patterns was assembled as input to our methodology. Via a focused literature review, all existing patterns relevant for our project were listed, i.e. patterns for systems that support the users' decision making through multi-modal and pro-active interfaces. Furthermore, new patterns were elicited in a bottom-up fashion, from the designs of the demonstrators discussed above.

Relevant patterns from literature

As a basis for our collection of patterns to be offered to the project consortium we explored the literature on multi-modal interaction design patterns. We selected, consulting [19], [17], [15], [13], [16], [18] and [20], 24 patterns deemed applicable to the type of systems envisaged in our project. To validate their applicability to our project, we organised a workshop among the designers of the different application domains. During this workshop, each of the demonstrator designers indicated which ones of the 24 identified patterns were relevant for their design. Each demonstrator identified about 13 relevant patterns. 10 patterns from the list were relevant for at least 4 of the 5 demonstrator designs.

Identifying new multi-modal patterns from designs

Method

As a method to identify new patterns, we used the 6 steps worked out by [21] on the basis of [8]'s motivation: observation, comparison, enrichment by imagination, prototyping, forces identification and applicability context analysis. During a workshop, the demonstrator designers were made aware of the concept and usefulness of patterns.

Figure 1: 'Context based information articulation' pattern in Emergency response

New patterns were identified in a combined fashion: HMI designers proposed a set of new patterns (top-down), demonstrator designers scrutinized their designs and proposed a set of patterns (bottom-up). Again in a very collaborative and iterative manner among the demonstrator designers of the 5 application domains, the proposed patterns were discussed, filtered and refined. However, not all proposed patterns were included in this final list since typically some system functions were confused with patterns. For example, route calculation or task management are functionalities, for which particular interaction patterns might be defined. The candidate list was filtered in terms of applicability to the different application domains, yielding a final list of 8 patterns:

- Multiple warnings
- Meaningful sounds
- Goal progression-based information detail and adaptation
- Mode switch
- Interactive instructions
- Modality combination for urgent messages
- Spatial point representation
- Context-based information articulation

Example

The following example illustrates the type of patterns that were newly identified:

Label	Description
Pattern name	Goal progression-based information detail and adaptation
Problem	The user is performing a task, while the goal and execution progression are known by the system. How to reduce information overload and provide the user with the essential and specifically tailored information?
Solution	At the appropriate time, the system adapts the level of information detail and perspective conveyed to the user, based on the user's proximity (in space and/or time) to her goal. In visual modality, this can be considered as a zoom into important information by refining the level of information detail and by providing complementary information. Consider using an extra modality in order to notify the user of a change during the task and display of information.

Table 1. Example of newly identified patterns

To present the patterns at a high level, we cite the label, problem and solution, and an example that gives a concrete idea of what the pattern intends to support. The full description of patterns will include worked out prototypes from the demonstrator designs.

Figure 1 illustrates the application of a new pattern in the emergency dispatching domain.

Pattern themes

Space limitation does not permit to list and define all patterns identified so far for our project. We have grouped the patterns into themes. The description of these themes gives an idea of the range and typology of all patterns relevant to our project:

System capacities - related to system features e.g. task management, commands…

Input - patterns that are either multi-modal, or specific to one modality (e.g. speech-enabled form) or two (e.g. multi-modal interactive maps)

Criticality support - applying to critical situations e.g. alerts, simulation…

Context-aware systems - requiring input about user state and context from sensing devices in order to adapt the interaction

Pro-active system behaviour - depending on the capacity of the system to anticipate/predict the needs of the user and react accordingly

Output - presenting information to the user in the most appropriate form

Although we were able to identify the patterns from the literature that are applicable to the designs and to describe new patterns in the same textual format, we observed that this level of description is not optimal in view of reusability in other domains.

Section 3 describes our endeavour to refine and formalize the description of patterns in a step towards a genuine design methodology.

PATTERN-BASED DESIGN METHODOLOGY

Goals

Re-use

The main motivation for describing existing solutions to known problems is to make this design knowledge available to others, both designers and non-designers. However, in order to be usable, patterns need to be described in a uniform way at the right level of detail so that they become applicable to other design contexts. Apart from a certain level of ambiguity associated with the textual description of the pattern problem and solution, the questions 'when?' and 'how?' also need to be answered. This information is missing from most multi-modal pattern descriptions we found in the literature, those found in [15] and [13] being noticeable exceptions.

Selection

The designer needs to be able to select the pattern that is applicable, based on the characteristics of a particular situation: context, information type and user type. Most pattern collections use a hierarchical classification scheme to support this process. Patterns belong to categories, possibly to subcategories, as in [15]. For instance, within the pattern category 'tables' one will find subcategories such as 'basic table', 'table with grouped rows', etc. The user's selection process is guided by the labels of the (sub)categories. This principle is also used in on-line pattern libraries, such as Welie.com [11] or Yahoo.com [10]. However, the limits of hierarchical classification are well known: different persons may classify elements in different classes, resulting in the difficulty to find an item. Moreover, a strictly hierarchical organization does not suffice to enable a motivated selection because it approaches the patterns from only one perspective. We need a richer organization where patterns can be selected from multiple perspectives and all their characteristics or attributes are taken into account.

The above described selection process applies to the selection at design time (i.e. when the interaction is designed, before implementation). However, in the applications that we envisage, some decisions need to be made at runtime (i.e. while the user is interacting with the system). For instance, depending on the context or the user state, a particular modality is not available. In this case, part of the pattern may be instantiated slightly differently. We might also envisage that an adaptive interface applies one or another pattern at runtime, depending on the user's context.

In the following sections we describe the methods incorporated in our methodology to match the above goals.

Methods

Structural description

In order to document patterns in a consistent way, we propose a structure that contains the necessary information to allow for re-use of this pattern in new designs. It is based on a proposal by [21] compiled on the basis of pattern structures by different authors, building further on the first attempts by [8]. This structure still needs to be validated by applying it to all our newly identified patterns. It describes the following pattern characteristics: name, task and problem, context, applicability, solution, consequences, examples and relationships.

Modelling framework

To meet all our goals however, we need a more formal level of analysis of the characteristics that specify a pattern. [22] and [23] have argued that, in order to be usable in practice, HMI design theory (guidelines) and expert knowledge (expertise and patterns) need to be formalised. They propose an ontology-driven semantic modelling framework and distinguish 3 levels of modelling abstraction: domain, expert and application-specific knowledge. Domain knowledge captures the generic HMI design knowledge in the form of concepts that are typical in multiple smart environments. For instance, the concepts 'user', 'device', 'primary activity', 'modality' are connected to each other by means of relationships ('uses', 'performs', 'supports_modality'). Expert knowledge incorporates the specific knowledge on multi-modal interaction residing in the community of expert designers (e.g. in the form of design patterns). Finally, application-specific knowledge formalises what needs to be known about a particular application area (in our case: the 5 demonstrator domains). For instance, it will be specified that the activity 'measuring_air_quality' in the emergency response domain requires the use of 'both_hands' and that the location 'emergency_site' 'has_noise_level' 'loud'. By separating these layers, a maximum reusability of design knowledge (i.e. design patterns) is guaranteed.

Moreover, [22] illustrates with an example that the formalisation of the features that determine the situation at hand (as captured e.g. by sensors) and the characteristics of the interaction between the user and the system allows for making decisions at runtime on the appropriate modality in a particular situation.

In line with this framework, we have derived a set of parameters to specify design patterns. Our hypothesis is that these parameters will facilitate the description of patterns, the selection of the appropriate one (at design time) and the selection of the right modality and other features of the interaction (at runtime).

<u>Parameters</u>

A set of parameters to be used for HMI pattern description and specification has been derived through multiple interactive and iterative sessions between a mixed team of HMI designers and ontology engineers. The goal of these sessions was to develop a uniform HMI pattern model reflecting the need for a formalized description and an increased level of detail, in order to be able to decide on pattern applicability in a certain design context. In the spirit of ontology-based modelling, the resulting model is a hierarchical class structure, describing pattern parameters and their attributes. Two major types of parameters have been identified: 1) parameters that characterize the pattern (see Figure 2) and 2) parameters that characterize the situation in which the pattern will be applied. The pattern parameters specify the essence of the pattern and impose constraints on the situation in order to be applicable. For instance, if a pattern is specified to have as an interaction mode 'visual output', the visual output interaction channel of the user needs to be available. On the other hand, some features of the situation might determine the instantiation of particular pattern variables (e.g. whether or not to use a particular modality).

Our study showed that the specificities of patterns can be described in a sufficient detail by means of the following 3 parameters:

Information: features of the information conveyed via the interaction e.g. type, volume to be presented, complexity

Interaction mode: characteristics and constraints of the interaction channel to be used e.g. modality and direction.

Interaction events: features of the interaction events triggered by the pattern e.g. what is its priority and importance for the user, whether the interaction will interrupt the task that the user is currently performing, does it require coordination with other events or tasks?

It also emerged from our study that the following 3 parameters are sufficiently expressive to fully characterise the situation in which patterns can be applied:

User: profile of the user, both intrinsic (seniority w.r.t. task, familiarity with the system) and context related (focus w.r.t. device, alertness, availability of senses, role in the interaction).

Context: the situation in which the user finds herself e.g. safety, stability, privacy, visibility and noise. Both physical and social context are modelled.

Device: features of the device(s) with which the user is interacting e.g. mobility constraints, size, previous interaction mode, modality, direction (in or output).

Various relationships have been derived between the parameter classes and subclasses as for instance: Device supports Interaction Mode, User executes Task, Task is PartOf Workflow, User has Context, Device sends Information, User interactsVia Interaction Channel, etc.

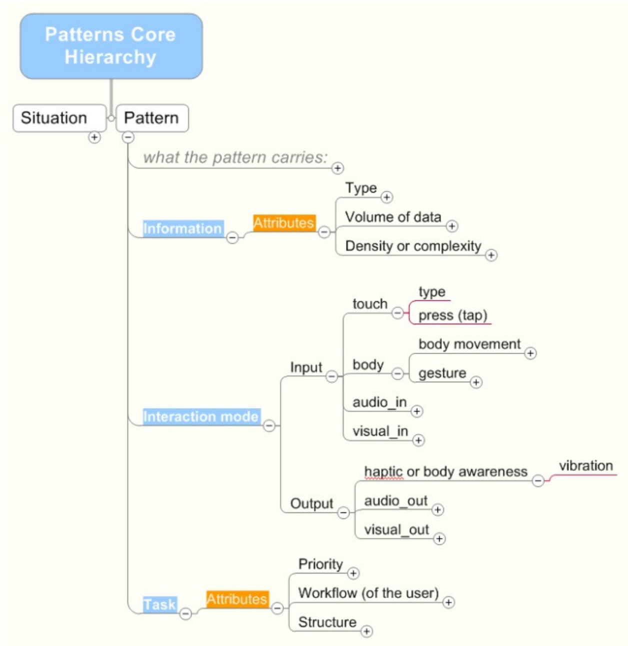

Figure 2: Semantic parameters that specify the pattern

CONCLUSIONS AND FUTURE WORK

The methodology set out in this paper lays the foundation for future work on several fronts.

First, the proposed patterns will be validated both at design time and at runtime. Most of the patterns will be instantiated in at least 2 of the 5 demonstrators of our project, hence we will find out which patterns are valid across domains. Feedback of the designers about the use of the patterns in the designs will allow us to refine, tune and enhance the patterns. Also, as the demonstrator prototypes have been developed, extensive user testing will yield additional feedback on the utility of the pattern to enhance the user experience.

Second, we will further elaborate the structural description formalism. Starting with the existing structure, we will refine it for multi-modal patterns. We will further investigate what are the best means to describe and visualize the specificities of multi-modal patterns.

The proposed hierarchical parameter model and the relationships defined between the different parameter classes are presently being implemented as a full-fledged ontology enabling further model refinement and reasoning.

Within the consortium, a runtime adaptive HMI engine is being developed on the basis of the parameters proposed in this paper. As described above, context values will be derived from sensor data and reasoning about these context values will enable to determine the applicability of a pattern in a particular situation or the selection of a particular variant of the pattern. Similarly, smart objects could behave differently according to the context by applying one or another interaction pattern at runtime.

Through this work, we aim at demonstrating that the modelling of appropriate data about user state and context linked to the specification of interaction patterns constitute a powerful means to realise a proactive, adaptive multi-modal interaction.

The benefits of this approach have already been demonstrated within the project consortium, as the patterns have fostered multiple communication exchanges about the similarities and the differences across the different demonstrator domains, in view of better grasping the essence of the envisaged type of systems. We believe that the results of this project, including the release of a well-documented collection of interaction design patterns, could benefit a larger community, encompassing the smart objects community, in particular but not limited to those involving interaction with users. This can be achieved by setting up a collaborative environment where different stakeholders in the design domain could interact, exchange ideas, improve and annotate patterns proposed by others and contribute new patterns.

ACKNOWLEDGEMENTS

This research is supported by the ARTEMIS JU (http://www.artemis-ju.eu/) and by the Brussels institute for research and innovation Innoviris (www.innoviris.be) through the ASTUTE project (www.astute-project.eu).

REFERENCES

1. Hudson, W. „Adopting User-Centered Design within an Agile Process: A conversation," *Cutter IT Journal,* vol. 16, nr. Software Usabillity, 2003.

2. Schaffer, E. Institutionalization of Usability: A Step-By-Step Guide, Addison-Wesley Professional, 2004.

3. I. International Organization for Standardization, „ISO 9241-16:1999 Ergonomics of human-system interaction," TC 159/SC 4 Ergonomics of human-system interaction, 1999.

4. Plass-Oude Bos, D., Poel, M., Nijholt, A. „A Study in User-Centered Design and Evaluation of Mental Tasks for BCI," *Lecture Notes in Computer Science,* vol. Volume 6524/2011, pp. 122-134, 2011.

5. Cooper, A. About Face: The Essentials of Interaction Design, IDG Books, 1995.

6. Goodwin, K., Cooper, A. Designing for the Digital Ag How to Create Human-Centered Products and Service Wiley, 2009.

7. Browne, D. P. Studio: structured user-interface design for interaction optimisation., Prentice-hall, 1994.

8. Alexander, C., Ishikawa, S., Silverstein, S. A Pattern Language, NY: Oxford University Press, 1977.

9. Oy, P. F. "Patterny," Pattern Factory Oy (Ltd.), [Online]. Available: http://patternry.com/. [Accessed 10 05 2012].

10. "Yahoo! Design Pattern Library," [Online]. Available: http://developer.yahoo.com/ypatterns/. [Accessed 10 05 2012].

11. Van Welie, M. "Patterns in interaction design - Patterns library," 2008. [Online]. Available: http://www.welie.com/patterns/. [Accessed 10 05 2012].

12. Morville, J., Callender, P. Search Patterns: Design for Discovery, O'Reilly Media, 2010.

13. Saffer, D. Designing Gestural interfaces, Sebastopol: O'Reilly Media Inc, 2009.

14. Scott, T., Neil, B. Designing Web Interfaces, Principles and Patterns for Rich Interaction., O'Reilly Media, 2009.

15. Tidwell, J. Designing interfaces, Patterns for effective interaction design, O'Reilly Media, 2010.

16. Godet-Bar, G., Dupuy-Chessa, S., Nigay, L. "Towards a system of Patterns for the design of Multimodal Interfaces," in *Computer-Aided Design of User Interfaces V, Proceedings of the Sixth International Conference on Computer-Aided Design of User Interfaces CADUI '06*, Bucharest, 2006.

17. Wolff, C., Ratzka, A. "A Pattern-Based Methodology for Multimodal Interaction Design," in *Lecture Notes in Computer Science, 2006, Volume 4188/2006*, Berlin, Springer-Verlag Berlin Heidelberg, 2006, pp. 677-686.

18. Ratzka, A. "Design Patterns in the Context of Multi-modal Interaction," in *Proceedings of the Sixth Nordic Conference on Pattern Languages of Programs*, Bergen, 2007.

19. Ratzka, A. "Identifying User Interface Patterns from Pertinent Multimodal Interaction Use Cases," in *Mensch & Computer 200*, München: Oldenbourg Verlag, 2008.

20. Mahemoff, M. „Design Reuse in HCI and Software Engineering, Phd Thesis," University of Melbourne, Depatrment of Computer Science and Software Engineering, Melbourne, 2001.

21. Kriwaczek, F. "Software Engineering Design II," 09 03 2009. [Online]. Available: http://www.doc.ic.ac.uk/~frk/frank/da/hci/pattern%20handout.pdf. [Accessed 10 05 2012].

22. Tourwé, T., Tsiporkova, E., González-Deleito, N., Hristoskova, A. „Ontology-driven Elicitation of Multimodal User Interface Design Recommendation," in *Proc. of The 23rd Benelux Conference on Artificial Intelligence, BNAIC 2011*, Ghent, Belgium, 2011.

23. Tsiporkova, E., Tourwé, T., González-Deleito, N. Hristoskova, A. „Ontology-driven Multimodal Interface Design for an Emergency Response Application," in *Proc. of the 9th International Conference on Information Systems for Crisis Response and Management, ISCRAM 2012*, Vancouver, Canada, 2012.