



HAL
open science

Supportive User Interfaces and Task Migratability in Smart Environments

Peter Forbrig, Michael Zaki, Philippe Palanque, Marco Winckler

► **To cite this version:**

Peter Forbrig, Michael Zaki, Philippe Palanque, Marco Winckler. Supportive User Interfaces and Task Migratability in Smart Environments. 3rd Workshop on Distributed User Interfaces: Models, Methods and Tools DUI @ EICS 2013, Jun 2013, London, United Kingdom. pp.42-45. hal-04083623

HAL Id: hal-04083623

<https://hal.science/hal-04083623>

Submitted on 27 Apr 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : <http://oatao.univ-toulouse.fr/>
Eprints ID : 12628

To cite this version : Forbrig, Peter and Zaki, Michael and Palanque, Philippe and Winckler, Marco Antonio *Supportive User Interfaces and Task Migratability in Smart Environments*. (2013) In: 3rd Workshop on Distributed User Interfaces: Models, Methods and Tools - EICS 2013, 24 June 2013 - 27 June 2013 (London, United Kingdom).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Supportive User Interfaces and Task Migratability in Smart Environments

Peter Forbrig, Michael Zaki
University of Rostock, Dep. of Comp. Science
Albert-Einstein-Str. 21
D-18051 Rostock, Germany
+49 381 498 7620
[peter.forbrig|michael.zaki]@uni-rostock.de

Philippe Palanque, Marco Winckler
ICS-IRIT, University Paul Sabatier,
118 route de Narbonne
31062 Toulouse CEDEX 9, France
+33 (0)5.61.55.63.59
[palanque|winckler]@irit.fr

ABSTRACT

This paper discusses supportive user interfaces as a special kind of distributed user interfaces. It introduces tangible objects as elements of a supportive user interface and discusses the role of metaphors. Furthermore, the role of supportive user interfaces for implementing the usability criteria of task migratability in smart environments is discussed. Some challenges are identified and the combination of tangible user interfaces and graphical user interfaces is suggested.

Keywords

Distributed user interfaces, supportive user interface, task migratability, smart environment

INTRODUCTION

The general term of “supportive user interfaces” fits to nearly all interactive applications as in some way every user interface has to be supportive. As a result of the SUI 2011 workshop participants agreed on the following more specific and precise definition:

“A supportive user interface (SUI) exchanges information about an interactive system with the user, and/or enables its modification, with the goal of improving the effectiveness and quality of the user's interaction with that system.” [7].

According to this definition a user interface should be distributed and adaptable in order to give the user the opportunity to interact with the system in a more appropriate way according to the specific encountered context of use. The idea of such interfaces is very much related to the “Meta-User Interface” approach [1], which has been introduced to control interactive ambient spaces.

A general definition of distributed user interfaces is given by Elmquist [5]. Additionally, five dimensions of such interfaces were identified. His definition states: “A distributed user interface is a user interface whose components are distributed across one or more of the

dimensions input, output, platform, space, and time.”

Our following discussion will be focused on distributed user interfaces in the context of smart environments. Some of these ideas were already presented in [10].

Existing approaches of supportive user interfaces will be discussed first. The usability concept of task migratability is presented afterwards and some aspects of tangible user interfaces are provided. Advantages and disadvantages of tangible user interfaces compared to graphical user interfaces are discussed and at the end we conclude our ideas of using tangible user interfaces as a special kind of SUI and to use SUIs for dynamic task allocation.

DISTRIBUTED USER INTERFACES IN SMART ENVIRONMENTS

Our experimental basis is a lab with installations of a smart meeting room that is equipped with a lot of sensors, projectors and cinema screens (see Figure 1). The room itself can be considered as a composition of distributed user interfaces. Bayesian algorithms try to infer next possible actions of the users and based on that information convenient assistance is to be provided.



Figure 1: Smart Meeting Room

The authors in [12] refer to the ambiguity of those systems by stating “This creates complex and unpredictable interactive computing environments that are hard to understand. Users thus have difficulties to build up their mental model of such interactive systems. To address this issue users need the opportunity to evaluate the state of these systems and to adapt them according to their needs.”

In this paper, Meta-UIs are suggested as a possible solution for the described problem.

A functional model and system architecture for Meta-User Interfaces for smart environments is provided by Roscher et al. [17]. The discussion focuses on the development of user interfaces that are distributed on different devices. "The Migration menu provides possibilities to redistribute a UUI (ubiquitous user interface) from one interaction resource to another, e.g. transfer the graphical UI to a screen better viewable from the users' current position. Through the Distribution menu the user can control the distribution on more fine grained levels by distributing selected parts of the UI among the available IRs."

In [18] the five features shapeability, distribution, multimodality, shareability and mergeability were specified. These results come originally from [1].

"1. **Shapeability**: Identifies the capability of a UI to provide multiple representations suitable for different contexts of use on a single interaction resource.

2. **Distribution**: Identifies the capability of a UI to present information simultaneously on multiple interaction resources, connected to different interaction devices.

3. **Multimodality**: Identifies the capability of the UI to support more than one modality.

4. **Shareability**: Denotes the capability of a UI to be used by more than one user (simultaneously or sequential) while sharing (partial) application data and (partial) interaction state.

5. **Mergeability**: Denotes the capability of a UI to be combined either partly or completely with another UI to create combined views and input possibilities."

Distribution is sometimes also called migratory or migratability. Grolaux et al. discuss in [12] the migration of parts of distributed interfaces to different platforms and devices.

All these mentioned features characterize the technical properties of user interfaces in given ubiquitous computing environments. However, these features do not cover the aspect of usability that is especially important for intelligent environments. Our discussion will especially focus on the dynamic allocation of tasks (task migratability) to devices, platforms, systems and users. This aspect is a little bit more abstract than the distribution of user interfaces. It will be discussed how task migrateability can be influenced by distributed user interfaces that we call supportive user interfaces.

We will especially discuss the role of tangible Meta-UIs for this purpose. Objects become part of a distributed interface. Such a user interface can even consist of several physical objects in a 3D space.

TASK MIGRATABILITY

Task migratability is known to be one of the usability criteria of interactive systems. It specifies the possibility of transferring of control for tasks execution between user and

system. "It should be possible for the user or system to pass control of a task over to the other or promote the task from a completely internalized one to a shared and cooperative venture" [9].

A lot of applications provide only a static solution that does not allow migrating tasks from devices to users or the other way round. Software developers often decide during design time which task is to be allocated to which actor.

The Importance of Task Migratability

The work of the ten authors cited the most in HCI is studied in [13]. We counted the number of times where a particular HCI principle was proposed. Task migratability is ranked the fifth among all usability principles after multiplying the counted number by a weighting factor derived from the author citation frequency.

Usually, computer scientists tend to delegate the supportive tasks to softwares running on computers. However, it is not mandatory to employ a computerized technique for the sake of providing an optimal support to resident actors. In some cases, tasks have to be performed by humans. It is obvious then that "designers need to acquire a deeper understanding of what the tasks of the users might be in certain situations and how to support their achievement." [8] Software solutions have to offer a way enabling users themselves to influence the distribution of tasks. Supportive user interfaces can help reconfiguring the applications.

Allocation of Function and Authority Sharing

When designing a (partly-) autonomous system, 10 levels of automation can be considered according to [16]. Work on function allocation such as the ones described in [11] or [3] aim at supporting the design of automation and more precisely at identifying and assessing candidate functions to be automated. Beyond that, if the use of the system is highly dynamic i.e. evolves regularly (for instance in order to handle unexpected adverse events such as malfunctions, faults, malicious attacks ...), there is a need for dedicated support to anticipating evolutions and for providing adequate solutions. Papers such as [1] propose a model-based tool-supported approach for the design and development of distributed user interfaces in the context of highly dynamic complex systems requiring repetitive and systematic activities to be allocated to the system in order to allow operators to be focusing on more analysis and decision related tasks. Such approach embeds automatic generation of distributed user interfaces allowing operators to monitor the execution of semi-autonomous procedures. Addressing function allocation correctly might have a huge impact on the usability especially in smart environment when quite often automation is discovered through usage. Bad designs might lead to so-called automation surprises having potentially disastrous impact on usability and possibly safety [15].

Task Migratability in Smart Environments

From our point of view task migratability is especially important within the domain of smart environments.

However, at the moment this has not been sufficiently discussed in the literature. The main tenet of smart environments systems is to support users as much as possible. Sometimes such systems are detailed designed and implemented for specific applications. However, most of the reported approaches work with Bayesian techniques that need training data. These data are provided by sensors reflecting the physical changes within an environment.

There are approaches like [17] that allow users to configure the environment. However, the concept of Meta-UIs is not directly related to task migratability between users and systems. Tasks are often only migrated between different devices. In this way the allocation of tasks between users and system remains the same. However, this technology can of course be used to influence task allocations to users as well. Supportive user interfaces for smart environments could be a solution. We strongly believe that this approach can improve the usability of smart environments. Preliminary experiments within smart meeting rooms support this hypothesis. However further experiments have to be performed to thoroughly evaluate the approach.

TANGIBLE USER INTERFACES

The idea of tangible user interfaces is not new. There is an ongoing research to use such interfaces in conjunction with table top systems. There is the intention to couple the different worlds of our reality as 3D space and the world of the computer. In [14] it is called coupling of bits and atoms.

In a smart ambience, the whole environment can be considered as a tangible user interface. Moving objects around like putting a notebook on a table can change the state of the environment's software. In the described example, a projector could connect to the notebook and present the slides on the computer. Even moving around as a human can be considered as an interaction. In our example with the notebook, the slides of that person standing in the room's presentation zone can be presented. In case this person leaves and another one enters the zone, the room acts then accordingly.

Our smart meeting room lab has an explicit tangible user interface. The tangible object is a pencil that is used in conjunction with a white board and a cinema screen. The white board is not visible when the screen is down. If the pencil is in a box, no white board is needed and the cinema screen can be down and be used by a projector. When the pencil is taken out of the box the screen has to go up to make the white board accessible. In case a projector was presenting on the screen, this projector has to be shut down.

The behavior is triggered by sensors in the box. However, from the point of view of the user, the pencil seems to be responsible for the behavior. The pencil however represents the tangible user interface and allows interacting with the environment. If the whole room is considered as a user interface for the smart environment, the pencil can then be seen as a very simple supportive user interface as it allows configuring the smart meeting room.

From our point of view, it may be very beneficial to consider real objects as parts of a distributed user interface, so that they can play the role of supportive user interfaces. Real physical objects can be used to express the kind of support that is appreciated by the users. In that way, explicit support is given by the users to the system to identify the current situation in detail. It might be difficult for the meeting room to identify whether a brainstorming meeting, a conference or a business meeting is taking place in the room. Even some sub-states such as meeting where full support, medium support or no support is required can be determined.

For this purpose a tea pot can be used. When it is placed on the meeting table, a brainstorming meeting is performed with a minor support. If there are two tea pots on the table some more is appreciated. Three tea pots specify that the maximum available support should be provided.

A tea pot placed on the side board signals that a conference is taking place in the room. If a tea pot is placed on a small table, a business meeting then is in progress.

In this case, the tea pots play the role of the supportive user interface. Their location configures the provided kind and level of support.

The consideration of real objects as distributed user interfaces and the concept of task migratability raise a lot of interesting questions.

1. Is it possible to decide in which situations it is better to use existing objects in the environment or to use new ones?
 - Pro existing: More convenient
 - Con existing: Objects might be used to drink tea or in order to reconfigure the system
 - Pro new: Less probability of unintended usage
2. Is it better to use one object at different locations or different objects at one location?
 - Pro one: More convenient
 - Con one: Difficult to remember
 - Pro several: Better to remember
 - Con several: Too many objects
3. Should existing metaphors be favored or new ones introduced?
 - Pro existing: Better learnability
 - Con existing: Confusion with traditional usage
 - Pro new: Clear interaction
 - Con new: Difficult to learn

There will be no general answers to those questions. However it would be interesting to discuss pros and cons with participants of the workshop and to identify some criteria that help to analyze application domains

TANGIBLE UIs VERSUS GUIs

It is of course not possible to express all necessary information that should be provided to an environment by tangible user interfaces. They are helpful only if a limited state space exists and one of these states has to be specified. The level of support or the limited number of presentation styles can be expressed easily.

However, it is not obvious when this is really convenient.

We identified different patterns for presentations like presentation with one projector, presentation in a sliding window mode with all projectors, presentation of the outline with one projector and presentation of the current slide with another one, presentation of the outline with one projector and a skidding window with three other ones, etc.

This can be expressed by a tangible user interface for users often working with the room. However, somebody new to the room would like to have a GUI to express the pattern for the presentation. Sometimes a certain file has to be selected for presentation. For this purpose tangible user interfaces are not helpful. However, if certain profiles are predefined the selection can be done in a tangible way.

A really broad variety of support is possible. Everything is configured with the help of a GUI like in [17] and [18], predefined configurations are selected by tangible objects or there is a solution using both ideas of a GUI and a tangible user interface. It seems to be really important to study the different approaches in detail and to consider also providing different kinds of support in parallel. In this way users can decide which kind of interaction they prefer. However, the unintended use of objects remains problematic.

Experiments will provide more insights for this interesting topic. However, we already truly believe that task migratability is really an important success factor for smart environments.

CONCLUSIONS

In this paper we argued to consider task migratability as an important aspect for smart environments. At the moment this aspect plays only a minor role in the scientific discussions concerning the implementation of smart environments. We believe that the acceptance of the concept of smart environments may increase if the user is able to influence the dynamic task allocation in a convenient way.

The paper introduced tangible user interfaces for smart environments as a special distributed user interface and a subset of supportive user interfaces. Tangible user interfaces can help to explicitly inform the environment about the current intentions of the user. Thus, implicit interactions in smart environments have an explicit influence. Also, future experiments need to show that the combination of tangible and graphical user interfaces can improve the usability of smart environments. In both cases task migratability has to be taken more into account.

During the workshop we would like to provide some examples for dynamic tasks allocation in smart meeting rooms and discuss their usefulness with the participants. We hope to get feedback from the participants concerning the design of tangible and graphical user interfaces.

REFERENCES

1. Barboni, E., Martinie C., Navarre, D., Palanque, P., Winckler, M. Bridging the Gap between a Behavioural Formal Description

- Technique and User Interface Description Language: Enhancing ICO with a Graphical User Interface Markup Language. *Journal of Science of Computer Programming* Vol. 78, 2013. (to appear).
2. Blumendorf, M. *Multimodal Interaction in Smart Environments A Model-based Runtime System for Ubiquitous User Interfaces*. Dissertation, Technische Universität Berlin, 2009.
3. Boy G. *Cognitive Function Analysis for Human-Centered Automation of Safety-Critical Systems*. Proc. of ACM SIGCHI conference on Human Factors for Computing Systems 1998: 265-272
4. Coutaz, J. *Meta-User Interfaces for Ambient Spaces*. In Proc. of the 5th Int. Ws. on Task Models and Diagrams for Users Interface Design: TAMODIA 2006, pp 1-15, Springer LNCS 4385.
5. Elmqvist, N.: *Distributed User Interfaces: State of the Art*, Proc. DUI 2011 CHI workshop, p. 7 -12.
6. Demeure, A., Lehmann, G., Petit, M. and Calvary, G. (Eds): *Proceedings of the 1st International Workshop on Supportive User Interfaces: SUI 2011 Pisa, Italy, June 13, 2011*, <http://ceur-ws.org/Vol-828/>.
7. Demeure, A., Lehmann, G., Petit, M. and Calvary, G. *SUI 2011 Workshop Summary Poster*, in [5]
8. Dittmar, A. and Forbrig, P., *Selective modeling to support task migratability of interactive artifacts*, Proceedings of the 13th IFIP TC 13 international conference on Human-computer interaction – Interact 2011, Lisbon, Portugal, Springer-Verlag, Volume Part III, ISBN 978-3-642-23764-5, p. 571-588.
9. Dix, A., Finlay, J.E., Abowd, G.D. and Beale, B.: *Human-Computer Interaction*, 3rd edn, Prentice-Hall, Englewood Cliffs (2003)
10. Forbrig, P.: *Interactions in Smart Environments and the Importance of Modelling*, *Romanian Journal of Human - Computer Interaction* Vol. 5 (2012) p. 1-12.
11. Fröberg, A., Eriksson, H., Berglund, E. *Developing a DUI Based Operator Control Station: A Case Study of the Marve Framework*. . In J.A. Gallud et al. (eds), *Distributed User Interfaces: Designing Interfaces for the Distributed Ecosystem*, *Human-Computer Interaction Series*, pages 1-12, 2011, Springer-Verlag, 2011.
12. Grolaux, D., P. V. Roy, P. V. and Vanderdonck, J. *Migratable user interfaces: Beyond migratory interfaces*. In *IEEE/ACM Conf. on Mobile and Ubiquitous Systems*, 422–430, 2004.
13. Hinze-Hoare, V. *Review and Analysis of Human Computer Interaction (HCI) Principles*, July 2007, <http://arxiv.org/ftp/arxiv/papers/0707/0707.3638.pdf>
14. Ishii, H. and Ulmer, B., *Tangible bits: towards seamless interfaces between people, bits, and atoms*. In: *Proceedings of the CHI'97 conference on human factors in computing systems*, Atlanta, Georgia, March 1997, pp 234 - 241
15. Palmer, E. "Oops, it didn't arm." - A Case Study of Two Automation Surprises . 8th Int. Symp. on Aviation Psychology, 1995.
16. Parasuraman, R.; Sheridan, T.B.; Wickens, C.D. "A model for types and levels of human interaction with automation" *Systems, Man and Cybernetics, Part A: Systems and Humans*, *IEEE Trans. on*, vol.30, no.3, pp.286-297, May 2000
17. Roscher, G., Blumendorf, M. and Albayrak, S. *Using Meta User Interfaces to Control Multimodal Interaction in Smart Environments*, *Proceedings of the IUI'09 Workshop on Model Driven Development of Advanced User Interfaces*, <http://ceur-ws.org/Vol-439/paper4.pdf>
18. Roscher, D., Lehmann, G., Blumendorf, M. and Albayrak, S. *Design and Implementation of Meta User Interfaces for Interaction in Smart Environments*, in [5]
19. Zaki, M., Wurdel, M., and Forbrig, P., *Pattern Driven Task Model Refinement*, *International Symposium on Distributed Computing and Artificial Intelligence, DCAI 2011, Advances in Soft Computing*, Vol. 91, ISBN = 978-3-642-19933-2 , p. 249-256.