MODELING THE TASK
Leveraging Knowledge-in-the-head at Design Time

George Christou
Department of Computer Science, Cyprus College, Nicosia, Cyprus

Robert J. K. Jacob
Department of ComputerScience, Tufts University, Medford, MA, USA

Pericles Leng Cheng
Department of ComputerScience, Cyprus College, Nicosia, Cyprus

Keywords: Human Computer Interaction, Design modeling, Mental model, Interaction Design, Evaluation framework.

Abstract: A key problem in Human Computer Interaction is the evaluation and comparison of tasks that are designed in different interaction styles. A closely related problem is how to create a model of the task that allows this comparison. This paper tries to tackle these two questions. It initially presents a structure (Specific User Knowledge Representation) that allows the creation of task models which allow direct comparisons between different interaction styles. The model allows the researcher or the designer to evaluate an interaction design very early in the design process.

1 INTRODUCTION

The revolution that is witnessed in interface design today, brings an impressive and diverse set of interaction styles, like Tangible User Interfaces (TUI) (Ishii & Ullmer, 1997), and many others. All these new interaction styles are becoming more varied and much less unified than previous generations, seemingly without cohesion on which to allow any modeling These Reality Based Interaction (RBI) (Jacob, 2004) styles are trying to mimic real-world manipulations, and draw from the skills that users already possess in the real world to allow the user to interact with the computer.

Because of this disparity, it is very difficult to characterize them and understand their underlying principles, like it was done for Direct Manipulation (Hutchins, Hollan & Norman, 1986). We believe though, that under this seeming disparity, there are many similarities both in the theoretical bases and the design approaches of RBIs.

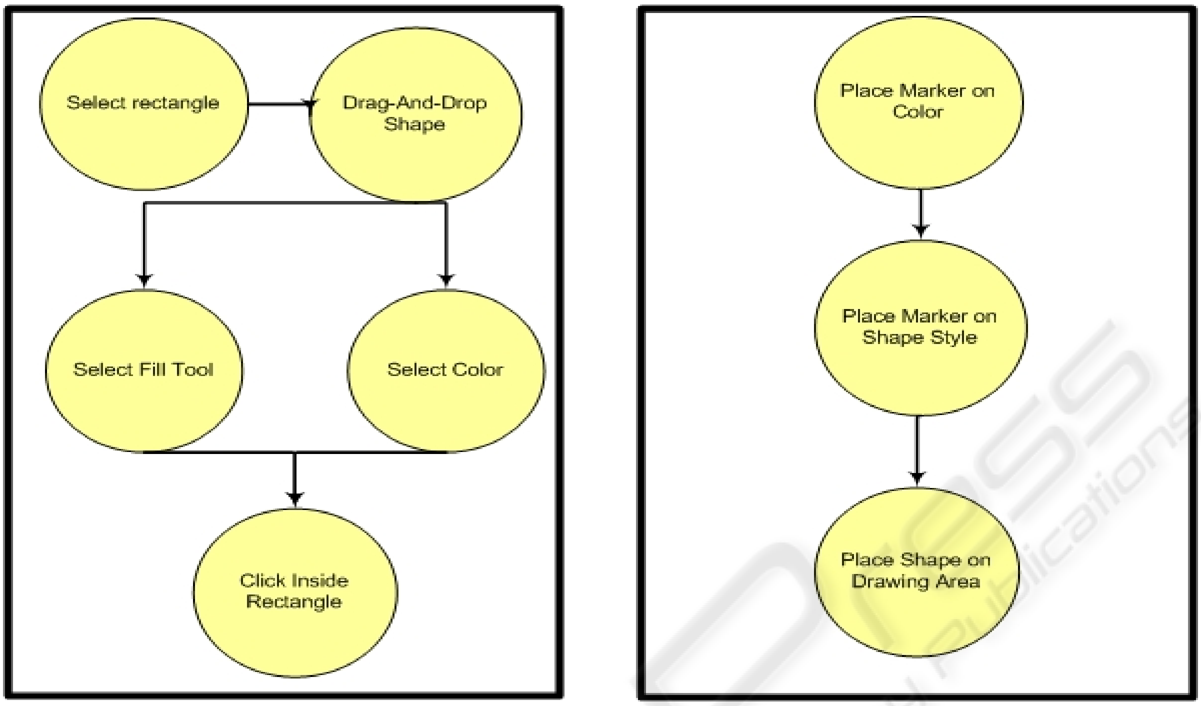
This paper presents a descriptive structure for the knowledge that RBIs leverage, and for their specification. The theoretical structure is called Specific User Knowledge Representation (SUKR) and it allows the modeling of user "knowledge-in-the-head" and interface "knowledge-in-the-world.

2 BACKGROUND

2.1 Theoretical Basis

The work presented in this paper is based on many different theoretical approaches. The basis of the research for the creation of the model though, is Task Analysis (TA), and more specifically, Cognitive Task Analysis (CTA). Diaper (2004) defines TA as "the collective noun used in the field of ergonomics, which includes HCI, for all the methods of collecting, classifying and interpreting

Christou G., J. K. Jacob R. and Leng Cheng P. (2006).
MODELING THE TASK - Leveraging Knowledge-in-the-head at Design Time.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - HCI*, pages 131-134
DOI: 10.5220/0002492301310134



(a) (b)

Figure 1: Task Diagram of creating a blue-filled rectangle in MS Paint™ (a) and TUIDraw (b).

data on the performance of systems that include at least one person as a system component" (Diaper, 2004, p.14). CTA is defined by Chipman, Schraagen and Shalin (2000) as "the extension of traditional task analysis techniques to yield information about the knowledge, thought processes and goal structures that underlie observable task performance" (Chipman, Schraagen & Shalin, 2000, p. 3). CTA theories provide specific methodologies for gathering and analyzing the appropriate data. They begin with a study of the jobs involved in order to determine which tasks should be analyzed (Chipman et al., 2000). The second step in CTA is to identify the knowledge representations that need to be used (Chipman et al., 2000). The final step is to use knowledge elicitation techniques that apply, based on the CTA theory of choice, since there exist many (Diaper & Stanton, 2004)

2.2 Task Knowledge Structures

Task Knowledge Structures (TKS) (Johnson & Johnson, 1991) is a theoretical and methodological approach to modeling tasks. It is a method of CTA that assumes that when people learn declarative and procedural facts that pertain to the same topic, the knowledge is not stored as stand-alone facts. Rather knowledge is grouped in coherent wholes, so that it can be recalled and used as a unit. TKS includes not only knowledge about actions, but also about objects used to perform those actions. In this way it falls under the external cognition theory proposed Norman (1988) and later by Scaife and Rogers (1996) which is discussed in section 3. TKS were designed to be a tool for design generation. By modeling user knowledge a designer can use the theory to generate design solutions for interactive systems (Hamilton, Johnson and Johnson, 1998). The presented model is based on the same assumption of Johnson and Johnson (1991)

Hamilton, Johnson and Johnson (1998) talk about objects in TKS and hint at the affordances (Norman, 1988) of objects, the object roles are not explicitly defined in terms of a user interface, nor are the affordances and constraints of these objects included in TKS. The proposed structure extends TKS with the addition of this information as shown in section 3.

2.3 Terms

The model uses terminology that was first presented by Christou and Jacob (2005). The terminology was created to allow researchers and designers to refer to

Pre-Conditions
1. Mouse-to-Pointer Static Binding
2. Single Click Action
3. Drag-and-Drop Action

Task Performance
1. Single-click on Rectangle Button
    a. Rectangle-to-Mouse Dynamic Binding
    b. Drag-and-Drop in drawing area to draw shape
2. Single-click on Fill Tool
    a. Fill Tool-to-Mouse Dynamic Binding
3. Single-click on Fill Color
    a. Fill Tool-to-Color Dynamic Binding
4. Single-click inside Rectangle

Figure 2: The Rectangle-Drawing Task in WIMP.

various parts of an interface without resorting to interaction style specific terms. The three parts that were identified in a user interface are the following:

1. The artifacts that represent the data that can be manipulated. We call these the *Data Objects* (DOs). The DOs are not the actual data in the system. They are the interface's representation of data in groups that are understandable and identifiable by a user.
2. The artifacts which are perceived by the user to be the means of interacting with the DOs. We call these *Interaction Objects* (IOs). The IOs are many times, the means by which the interaction occurs between the user and the interface.
3. The actual artifacts that are manipulated by the user in order to manipulate the IOs. We call these the *Intermediary Objects* (INs). The intermediary objects are, most of the time, coupled with an IO, and this relationship is usually constant.

2.4 Bindings

Bindings are the places where the IO and the DO become connected, in order to carry out an action by the user. When an IO touches or in some other way comes connects with the DO, we say that the IO is bound to the DO.

Bindings reveal the places where interaction is possible between the user and the DOs. These places show where interaction is possible between the tools and the data of the interface. Since these are the major places where actions take place, it would be sensible to single them out for study.

Two types of bindings are identified. Static bindings, where the relationship between two objects is always there and never changes, and dynamic bindings, where the binding exists only for a limited time, usually until the user finishes an action or a task.

For example, the binding between the mouse and pointer is static, because it never changes, no matter what the application does. The binding between an icon and the pointer during a drag-and-drop action however, is dynamic, because it only exists during the time of the action.

Pre-Conditions
- Marker use.

Task Performance
1. Place marker on color
    a. Marker-to-color Dynamic Binding
2. Place marker on Filled Shape
    a. Marker-to-Filled Shape Dynamic Binding.
3. Place shape on drawing area.

Figure 3: Rectangle Drawing Task in Tangible User Interface.

3 SPECIFIC USER KNOWLEDGE REPRESENTATIONS

The model that results from the analysis of the interface is the Specific User Knowledge Representation (SUKR). SUKRs are a form of CTA, and are based on the theory of external cognition (Scaife & Rogers, 1996). Scaife & Rogers (1996) postulate that humans do not only use mental models and mental representations to interact with the world. They argue that artifacts in the real world are very much part of reflective behavior. When humans perform any cognitive task, they use artifacts that become part of the problem representation, and the correct user model of the artifact allows for better solutions to the pertinent tasks.

The model presented here is a task model and not a user model, thus it does not require any goal driven behavior. The model describes the task based on some interaction style, using the interaction style's actions, rather than the user's or researcher's model of how the task should be performed. The model tries to capture the knowledge needed for each task that can be performed in the interface under some interaction style.

SUKRs are comprised of two parts, each with a specific goal:

1. The pre-conditions section, which aims to capture the minimum amount of procedural and declarative (but not domain) knowledge needed by the user to perform the task in the given interface. The model supposes that domain knowledge is constant over all interaction styles, and
2. The task performance section, which describes the way the user should perform the action in terms of the DOs, IOs, and INs, and by including the necessary Bindings.

Domain knowledge is considered constant throughout interaction styles and that is why it is not considered in the SUKR.

The task used to show the modeling procedure is drawing a filled rectangle in MS Paint™ and in TUIDraw, a Tangible User Interface Drawing program, bult by the authors. Fig. 1 shows an example of a task in two interaction styles. From the task diagram, the SUKR may be created in the following way. The diagram is created by breaking the task up in its constituent actions and each action is represented by a circle. Any actions that may be performed in any order are signified by placing their circles in the same level of the diagram. For each action the knowledge needed to perform it is delineated.

The common knowledge for all tasks, such as the static binding of the mouse to the pointer and that left-clicking on buttons changes the function of the pointer is put in the preconditions section of the SUKR, and knowledge specific to the execution of the action, and dynamic bindings that occur during the execution of the actions of the task are put in the task performance section. The full SUKRs can be seen in figs. 2 and 3.

4 CONCLUSIONS AND FUTURE WORK

In this paper the concept of Specific User Knowledge Representations was presented, along with the relevant specification method.

Future work that needs to be done is to clarify and specify the procedure for knowledge elicitation, and more experiments need to be performed, mainly with experienced users, to show that the measure holds not only for novice performance, but also for intermediate and expert users.

REFERENCES

Chipman, S. F., Schraagen, J. M. & Shalin, V. L. 2000. Introduction to Cognitive Task Analysis. In Chipman, S. F., Schraagen, J. M. & Shalin, V. L. (Eds.) 2000. Cognitive Task Analysis. Mahwah, NJ: Lawrence Erlbaum Associates

Christou, G. and Jacob, R.J.K. 2003. Evaluating and Comparing Interaction Styles, Proc. of DSV-IS 2003 10th Eurographics Workshop on Design, Specification and Verification Interactive Systems. Springer-Verlag.

Christou, G. & Jacob, R. J. K. 2005. Identifiability: A Predictive Quantitative Measure for the Comparison of a Task Designed in Different Interaction Styles. Proceedings of HCI International 2005. Lawrence Erlbaum Associates. CD-ROM.

Diaper, D. 2004. Understanding Task Analysis for Human Computer Interaction. In Diaper, D. & Stanton, N. (Eds.) 2004. The Handbook of Task Analysis for Human-Computer Interaction. Mahwah, NJ: Lawrence Erlbaum Associates, pp. 5-47.

Diaper, D. & Stanton, N. 2004. The Handbook of Task Analysis for Human-Computer Interaction. Mahwah, NJ: Lawrence Erlbaum Associates.

Hamilton, F., Johnson, P., and Johnson, H. (1998) Task-Related Principles for User Interface Design. Van der Veer, G.C. (Ed). Proceedings of the Shaerding Workshop on Task Analysis, Austria

Hutchins, E., Hollan, J., & Norman, D. 1986. Direct Manipulation Interfaces. In D. A. Norman & S. W. Draper (Eds.) User Centered System Design: New Perspectives in Human-Computer Interaction. Lawrence Erlbaum Associates: Hillsdale, NJ.

Ishii, H. and Ullmer, B. 1997. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. Proc. ACM CHI'97 Conference on Human Factors in Computing Systems. pp. 234-241.

Jacob, R. J. K. 2004. Reality-Based Interaction: A New Framework for Understanding the Next Generation of Human-Computer Interfaces. Retrieved October 18, 2005, Tufts University, Dept.of Computer Science Web Site:http://www.eecs.tufts.edu/~jacob/ theory/

Johnson, H., and Johnson, P. 1991. Task Knowledge Structures: Psychological Basis and Integration into System Design. Acta Psychologica, 78 pp. 3-26.

Norman, D. 1988. The Psychology of Everyday Things. New York: NY. Basic Books.

Scaife, M. & Rogers, Y. 1996. External Cognition: How do Graphical Representations Work? International Journal of Human Computer Studies, 45, 185 – 213.