DIVISION OF COMPUTER SCIENCE

Integrating Formal and Diagrammatic Techniques in Requirements Capture and Early System Design

Jill Hewitt
Sara Jones
John Sapsford-Francis

Technical Report No.188

March 1994

Technical Report No. 188 March 1994

Integrating Formal and Diagrammatic Techniques in Requirements Capture and Early System Design

Jill Hewitt, Sara Jones and John Sapsford-Francis
comqjah, comrsj, comqjs @herts.ac.uk

School of Information Sciences, University of Hertfordshire,
College Lane, Hatfield, Herts, AL10 9AB, UK

Tel: (0707) 284766, 284327, 284370, 284354
Fax: (0707) 284303

Submitted for presentation at the Workshop on Research Issues in the Intersection Between Software Engineering and Human-Computer Interaction, held before the 16th International Conference on Software Engineering, Sorrento, May 1994.

Workshop on Research Issues in the Intersection Between Software Engineering and Human-Computer Interaction, Sorrento, 1994

Position Paper:

Integrating Formal and Diagrammatic Techniques in Requirements Capture and Early System Design

Jill Hewitt, Sara Jones, John Sapsford-Francis

J.Hewitt, S.Jones, J.Sapsford-Francis@herts.ac.uk
School of Information Sciences, University of Hertfordshire,
College Lane, Hatfield, Herts, AL10 9AB, UK

Introduction

The process of design in software engineering is a complex one. Decisions made in the design stage have far reaching, and sometimes devastating effects on subsequent development work. Adopting a user-centred approach to system development adds a further layer of complexity which must, in turn, be addressed. There is considerable experience at the University of Hertfordshire of developing novel human-computer systems through user-centred design (see, for example, [1] and [2]). This experience provides a strong focus for our interests in enhancing the quality of the development process by integrating formal methods with the less formal techniques currently used.

Background and Motivations

This paper describes our work on formative aspects of the design process. We take as our starting point representations used in the design of selected information and knowledge-based systems using multimedia interface technology. Examples of these representations are being collected and analysed, using interviews with their authors, and with others who must use them, to discover the significance of particular diagrams to each participant in the design discussion. We are currently investigating whether such informal representations of early design ideas can be used as an intermediary stage in the transition to more formal specifications of aspects of the human-computer interface. In connection with this, we are considering whether the introduction of greater precision into the semantics of diagrammatic representations might support better reasoning and communication, both within the design team and with users.

Initial system design is often a problem-solving activity which relies on knowledge based performance [3]. Designers use this kind of problem-solving approach to tackle novel problems through the use of analogical and heuristic reasoning and the application of generic problem solving approaches. They typically use a wide range of informal representations. In design meetings associated with ongoing projects at Hatfield, members of

design teams often use diagrams of various forms to communicate their ideas and form a basis for discussion. Our experience in the field of knowledge-based systems suggests that knowledge engineers also favour the use of graphical representations during the early stages of system design [4].

As design issues become better understood, there is a need for notations which allow requirements and proposed solutions to be specified with greater precision, and which therefore support more effective communication between designers and users. Software engineers are looking increasingly to the use of formal methods and notations to provide such precision. Studies have shown that using formal methods early in system design tends to reduce the overall cost and development effort for a project by forcing designers to confront inconsistencies and possible problems with a proposed design at an early stage [5]. In recent years, there has been increasing interest in the application of such methods in the process of human-computer interface design. At the University of Hertfordshire, we have begun to investigate the use of state-based formal notations such as Z to specify properties of graphical interface components [6], and process-based formalisms such as CSP to describe dialogues with hypermedia information systems [7]. We aim in future to investigate the role of proof in the formal specification of elements of the human-computer interface and in verification of user requirements.

A focus of interest for us now is the way in which informal diagrammatic descriptions of various aspects of a system design might be linked to more formal specifications of those properties in a way which will permit interface designers to derive benefits of formal methods such as those envisaged by software engineers. The diagrams commonly used by designers to help them to reason about particular system designs are often unstructured and do not use any one set of drawing conventions. The level of detail often varies in different parts of a diagram, depending on the difficulty (to the designers) of describing the corresponding parts of the system. Designers may also introduce new conventions to describe problems encountered for the first time in new domains or with novel system architectures.

In a close-knit design team, the fluidity of diagram semantics may not cause a problem, with the designers communicating effectively, even through changing and informal specifications. However, in larger design teams whose members have various levels of experience, our study has shown that misunderstandings can easily arise. The backtracking from formal specifications or prototypes developed on the basis of such mistaken understandings is at best time-wasting, and can have more serious implications if customers have also been misled by the informal descriptions.

The following paragraphs present examples of diagrams used during the course of one meeting associated with the development of a multimedia information system at Hatfield and demonstrate that while some such diagrams may form a basis for formal specification, others may not.

Some Examples

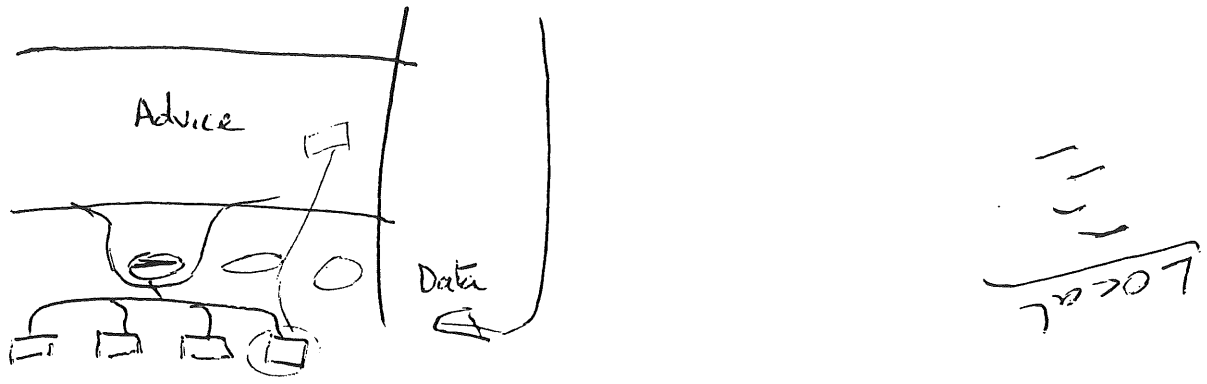

Figure 1: Informal diagrams drawn during a meeting discussing the design of a multimedia information system.

The diagrams shown above were drawn during a design meeting for the SPIRE project. SPIRE aims to develop a multimedia information system from which users will be able to obtain advice on integrating students with disabilities into higher education. The system is conceptualised as being made up of three 'layers' (see [8]). The top layer, the one most users will see first, presents a task-based view of the domain which is intended to guide users to information of the kind they need. After the task layer, users will typically be presented with information from the 'advice' layer which contains the kind of general expert advice currently provided by a range of support workers. Finally, the data layer contains basic items of information about, for example, particular pieces of equipment for students with disabilities, or facilities at the Health Centre. A decision to store data in the data layer using a relational database package and implement the other two layers using hypertext had already been made at the time of the meeting, but many aspects of the design had yet to be finalised.

The diagram on the left was drawn during a discussion of the way in which information presented to the user should be divided between the advice and data layers. The particular question of interest was whether descriptions of groups or classes of objects about which information was held in the data layer were general enough to be thought of as part of the advice layer. In interviews carried out after the meeting, there was found to be broad agreement about the issue to which the diagram related, and the meaning of its various components. This diagram would therefore form a good basis on which to construct a formal specification of relevant aspects of the system and interface.

The diagram on the right is obviously much sketchier than the other, and was the subject of less discussion. In interviews after the meeting, it was discovered that each member of the team had a different view of the significance of this diagram, and there was no agreement as to the design decision reached in connection with the issue under discussion. Designs constructed on the basis of the different views expressed might have been radically different. In this case, it would obviously have been dangerous for a particular member of the team to use the diagram as the basis for a formal specification.

Conclusions

We acknowledge that problems such as those described above can be partly solved by managerial measures: for example, by having someone, possibly a member of the project team, taking minutes for each meeting and recording design decisions as they are made. However, members of the project team are often too actively involved in discussion to be able to record all that is said, and those outside the team will often not have a proper understanding of the issues under discussion. In any case, the minutes of such a meeting would simply represent a single individual's view of what took place, which might well be different from those of others present, just as views about the significance of diagrams described above were different.

One solution we are considering involves using graphical notations with well-defined semantics (corresponding, at least loosely, with the semantics of relevant formal notations) to record decisions reached at the end of every meeting in a form which can immediately be validated and agreed upon by all present. We believe that informal diagrams used during the meeting might translate quite easily into more precise graphical notations. Diagrams drawn using such notations might be easier for the majority of designers to generate and validate than full formal specifications. An added advantage might be that such diagrams could be used to communicate with users at appropriate points in an iterative user-centred design approach. With these things in mind, we are beginning to investigate what forms of graphical representations might be most suitable for use in these ways.

References

[1] J. Hewitt, J. Sapsford-Francis and P. Halford. Use of Multi-Media in a Public Information System. In *Proceedings of the 1993 International Symposium on Multi-Media Technologies and Future Applications,* BCS/IEEE/RTS, Southampton, April 1993.

[2] J. Sapsford-Francis, J. Hewitt, P. Halford and N. Waring. Human Factors Design of a Multi-Media System: the Libtech International '93 Exhibition Guide. Presented at the Libtech International Exhibition. 1993.

[3] J. Reason. *Human Error.* Cambridge University Press, 1990.

[4] S. Jones. 3-d Diagrams for Knowledge Engineering: An Early Estimation of Utility. In *Human-Computer Interaction: Software and Hardware Interfaces,* G. Salvendy and M. J. Smith (eds), Advances in Human Factors/Ergonomics volume 19B, Elsevier, 1993. Proceedings of the Fifth International Conference in Human-Computer Interaction (HCI International '93), Orlando, Florida, August 8-13, 1993, volume 2.

[5] A. Hall. Seven myths of formal methods, in *IEEE Software* , September 1990.

[6] S. Jones. *Three-Dimensional Interactive Connection Diagrams for Knowledge Engineering.* PhD thesis, City University, 1993. Chapter 11.

[7] L. Jacob & S. Jones, *Formal Dialogue Specification for Hypertext and Multimedia Systems* , Technical Report No. 175, Computer Science Division, University of Hertfordshire, January 1994.

[8] M. Bearne, J. Hewitt, S. Jones & J. Sapsford-Francis, *An Integrated Approach to the Development of Advice Systems to Support Learning and Domain-Based Information Retrieval* , Technical Report No. 180, Computer Science Division, University of Hertfordshire, February 1994.