The Knowledge Engineering Review, Vol. 10:1, 1995, 63–66

Models and Techniques for the Reuse of Designs: ECAI94 Workshop, Amsterdam, August 8 1994

P. M. WOGNUM

University of Twente, Department of Computer Science, Knowledge-Based Systems Group, P.O. Box 217, 7500 AE Enschede, The Netherlands

1 Introduction

Reuse of designs is a topic which is receiving growing attention from the research community. Reuse of designs addresses the reuse of former experience in new design situations; relevant existing cases are retrieved and adapted to new situations. This reuse is common practice for engineers working in practice, both for efficiency reasons and to prevent making the same mistakes over and over again.

At the 11th European Conference on Artificial Intelligence, held in Amsterdam, August 1994, a one-day workshop was organized on models and techniques for the reuse of designs. The workshop focused on the problems that need to be addressed to offer automated support to the reuse of designs in real applications.

The workshop consisted of ten presentations which addressed interesting problems to be solved. The discussions among the 22 participants were lively, and contributed to the overall outcome of the workshop; an inventory of challenging research areas.

2 Background

The reuse of designs consists of two major aspects, retrieval and adaptation. Retrieval is a part of case-based reasoning, and consists of retrieving old design cases. Adaptation, also called redesign or modification, consists of changing an old design to satisfy new requirements. Research in the field has been performed for several years now, and has led to a number of results (Maher & Zhang, 1991; Kolodner, 1993; Goel & Chandrasekaran, 1992). However, despite the many interesting ideas described in the literature, the research has also revealed many problems which need to be solved before we may arrive at applications which are really useful in practice.

Reuse of designs is studied for three reasons (Tempelman et al., 1994). The first reason is the reuse of former cases for explanation. For instance, the use of an example to explain a particular solution technique. The second reason is the enhancement of the efficiency of a reasoning process, especially in knowledge-rich environments such as design. It is common knowledge that starting with a relevant case is often more efficient than starting a design process from scratch. The third reason is the reuse of designs to compensate for the lack of formal knowledge, which is especially true in the conceptual design phase.

As already indicated above, two major research areas can be distinguished. These areas are classification and modification. Classification involves modelling cases in suitable classes and retrieving them in new design situations. The classification of cases is determined for a large part by the similarity defined between a new problem and the existing cases. Questions in this area are: What features to use? What types of features to use? What is the relation between the context of a new problem and the feature to be used for retrieving a case? How specific or generic must features be? How complex are features? Which aspects need to be taken into account to assess similarity?

In the same vein, to be able to support modification a number of questions need to be addressed, such as which parts need to be modified, what aspects determine the best solution, or which methods can be used to adapt a design?

In the workshop some of the issues raised above were addressed. The discussions on these topics have led to an inventory of ongoing and future research areas. The results will be discussed below.

3 Results

The application areas represented by the participants of the workshop are software design and engineering design in several technical areas. As became clear in the workshop, the problems addressed are conceptually the same for both software design and engineering design. For the actual adaptation of a design, the differences between methods and techniques may become more apparent. In the workshop, however, the problems were mainly discussed and compared at the conceptual level.

The themes discussed in the workshop were the following:

3.1 Implicit knowledge

In modelling requirements, products and design decisions, much knowledge is often left implicit. This is not a serious problem if the use of these models is restricted to a particular type of applications in similar contexts. However, to allow for the reuse of these models in larger contexts and broader application areas, it is necessary to make the knowledge underlying the models explicit.

Requirement qualifications are one approach to modelling the assumptions underlying functional specifications. By challenging these qualifications, changes can be made in the functional specifications resulting in changes in the design (Wijngaards et al., Free University Amsterdam). On a more abstract level, making explicit the basic function a design must fulfil helps in finding former abstract solutions to new difficult design problems (Sushkov, University of Twente, The Netherlands/IMLab, Minsk). This approach is based on the notion that even creative or innovative design can be supported by reusing former experience (Altshueller, 1984).

Several assumptions are made in developing a model of a real artifact. Making these assumptions explicit will help in reusing and redesigning the models in different situations. For instance, an inventory and classification of assumptions underlying simulation models of dynamic systems will support the reuse of simulation models and their adaptation if they do not yet adequately explain the system's real behaviour (Pos et al., University of Twente/ECN, The Netherlands). Similarly, (software) components are often designed to perform one function in only one context, resulting in (implicit) functional fixedness. Challenging the assumptions underlying a particular solution may remove this functional fixedness. Removing the functional fixedness makes components reusable in larger contexts (Dusink, Technical University of Delft, The Netherlands).

During a design process, especially a process performed by a team of experts, a design proposal is often not accepted the first time. Improving such a design proposal involves much discussion and comparison of alternatives. The rationale behind design decisions, however, is often left implicit. Making this rationale explicit would considerably help in improving design proposals and in preventing mistakes being repeated (Jenkins, University of Paisley, UK). The discussions on this topic revealed, however, that capturing this knowledge is extremely difficult.

3.2 Similarity

Classical approaches to similarity do not take into account the meaning of the variables involved in the comparison of a new problem with old cases. Such approaches often lead to many false-positive or false-negative cases.

If the meaning of variables is taken into account in determining the similarity between cases, similarity measures can be considerably improved. This improvement has been shown by some experiments to determine the difference between similarity based on syntactic and semantic indices

(Maurer, University of Kaiserslautern, Germany). The semantics of the variables involved in these experiments include several aspects of a design description.

A second improvement in determining the usefulness of a case for reuse in a new situation can be achieved by combining retrieval and adaptation of the case (Smyth, Hitachi Dublin Laboratory, Ireland). By taking adaptation into account, more accurate retrieval is achieved and cheaper adaptation. Although the method suffers from a more complex retrieval process because adaptation must be predicted for each case considered, the costs will be repaid by a much cheaper adaptation process.

3.3 Diagnosis

It may not always be obvious which part of an existing design or a retrieved case must be adapted to meet the requirements in a new design situation. Parts designated at first glance to be suited for adaptation may appear less suited after closer inspection. Methods are needed to support the choice of the parts of a design that must be modified to satisfy the requirements.

One such method may be found in model-based reasoning. In particular, the methods that have been developed in model-based diagnosis research may be suited to find the causes for the discrepancies between the new requirements and the behaviour of the old case. A prerequisite for the application of such methods is that the design description must consist of components, connections between components, and behavioural descriptions of components. Application of one model-based diagnosis technique (van Eldonk, University of Twente, The Netherlands) showed that the approach is in principle possible. In addition, application of the model-based diagnosis technique results in new specifications for the part designated for adaptation. However, the method has to be extended to be useful for realistic systems. Moreover, the method results in more than one solution. No help is offered yet to determine which solution is best.

3.4 Reuse of design methods

Designers tend to reuse transformations of designs in new design situations (Dyke, Paisley University, UK). In experiments, seven such basic transformations were identified. Before more general design methods can be modelled and formalized to be reusable, however, understanding of the design process is needed. There exists no appropriate model of the design process yet which can be used for the purpose. An important aspect of design process models is the degree to which such models must take into account current (human) design practice.

3.5 Adaptation

This issue is considered to be the most difficult, but also a very important part of reuse of designs. An example of adaptation of design descriptions is the geometrical adaptation of building designs (Smith, AI Lab of the Federal Institute of Technology, Lausanne, Switzerland). In this approach, the integration of different views on a design has been made possible through a representation of requirements as mathematical constraints. Run-time parameterization of constraints helps in reducing the number of constraints to be considered for a specific problem. The approach has proven to be useful for dimensional adaptation. Additional structuring of the constraints in intelligent objects also enables topological adaptation.

4 Discussion

The overall result of the workshop is that we need to address the question "What are we using reuse for?" to be able to model the knowledge needed to offer automated support to the reuse of designs.

A distinction was made between similarity in routine design and in innovative design. For the first type of design, determining similarity might be fairly simple, because the changes needed to adapt a design to new requirements might fall within well-defined ranges. For the second type of design, finding a case relevant for the new design situation will be more difficult, because there is no simple similarity. Instead, to support reuse in innovative design, we have to find analogous cases. To accomplish this, we need more abstract descriptions of cases. It is still an open question as to what abstraction types we need for modelling cases for reuse in innovative situations.

It was stressed by several participants than an important part of automated support should assist in adapting a design case. To accomplish such support, models and techniques for several more specific tasks must be developed. Examples of such tasks are choosing the parts that need to be modified, estimation of the complexity of an adaptation and its consequences on the overall design, including the costs of adaptation, and finally, the actual adaptation. It is also an open question as to whether the models and techniques for supporting routine adaptations can be similar to, or must differ considerably from, those for supporting innovative design.

In summary, we need models and techniques for modelling, representing and using cases in both routine design and innovative design. Innovative design is often needed in the earliest phases of design, where later phases of a design process are usually characterized by a more routine nature. Routine design requires specific application-dependent knowledge, while innovative design requires more generic knowledge applicable in a wide range of design situations. We must study whether these types of knowledge can be modelled with generic design concepts at different abstraction levels, or require different modelling approaches.

Acknowledgements

I would like to thank the members of the organizing committee for their support in organizing the workshop. I also thank the participants of the workshop for their contribution to the workshop notes and the discussion which made the workshop a success. I want to mention in particular Professor Ernest Edmonds for closing the workshop in his own challenging way. Finally, I thank the two members of the ECAI94 workshop organizing committee for their assistance.

References

Altshuller, GS, 1984. *Creativity as an exact science: the theory of the solution of inventive problems.* Gordon and Breach.

Goel, A and Chandrasekaran, B, 1992. "Case-based design: a task analysis" In: *Artificial Intelligence in Engineering, vol. II* (C Tong and D Sriram, eds.). Academic Press, pp 165–183.

Kolodner, J, 1993. *Case-based Reasoning.* Morgan Kaufmann.

Maher, ML and Zhang, DM, 1991. "CADSYN: using case and decomposition knowledge for design synthesis" In: *Proceedings of AID91* (J Gero, ed.). Butterworth-Heinemann, pp 137–150.

Tempelman, F, Wognum, PM and Mars, NJI (1994). "Case-based reasoning in literature and Stevin". Technical Report UT-KBS-94-12, University of Twente, Enschede, The Netherlands.

Wognum, PM (Ed), 1994. "Workshop Notes of the ECAI94 Workshop on Models and Techniques for Reuse of Designs".