Chapter Z

Structured Methods and Creativity: a happy Dutch marriage

Martijn van Welie
Gerrit C. van der Veer

Z.1 Introduction

DUTCH is a design method for complex interactive systems. Such design projects require multidisciplinary teams, where the various members each have the responsibility for some aspect of design. This poses constraints on the use of documentation, representation and communication within the team and in relation to clients and users. The needed representations vary in character, from formalisms to scenarios and highly informal sketches.

From our experience in applying DUTCH in teams in educational settings and industry, we will show that the combination of representations does not limit creativity but, on the contrary, stimulates creative input from designers, clients, and prospective users. Informal and open scenarios are built on the basis of formal representations of task models and evaluation is done by systematic heuristics walkthroughs as well as by acting out the scenarios.

Designing complex interactive systems is rarely an individual activity. As soon as different user roles can be distinguished, and technological innovations affect roles, procedures, and other organisational aspects, various disciplines have to be involved, and these have to work in a structured design team. In industry, such large systems are typically designed in groups of 10 to 30 persons where subgroups are formed that handle particular activities. Our method DUTCH is intended for the development of such systems. We suggest allocating complementary activities to small specialised groups of around 3 to 5 persons. The groups have to work together to complete the design. Breaking up the work is necessary when such large projects are undertaken. When groups work together it puts constraints on the way communication is handled. We found that a combination of formal and informal representations is well suited to this kind of communication.

In the following sections we will highlight the kind of representations that we use in the various design activities. First we will give a short overview of our method DUTCH in order to explain the activities it comprises and their purposes.

Z.2 Dutch Design

Over the past years we have taken useful bits of theories and combined them into a coherent practical method for designing complex interactive systems. The resulting method is called DUTCH (Designing for Users and Tasks from Concepts to Handles). The method has been used successfully in both industry and education proving the practical value of the method. From experiences such as outlined in the example, we learned that for a practical method it is required to a) define a clear *process*, b) define the *models and representations* including their semantics and c) support the method and models with *tools*.

Our design process is task based which means that it uses the tasks of users as a driving force in the design process. The goals are to design both *usable* and *useful* systems. We think it is important to base the design on the *work* that has to be done by the users. Therefore, the users play an important role in acquiring knowledge about their work as well as for usability testing.

Our process consists of four main activities: (a) analysing a "current" task situation, (b) envisioning a future task situation for which information technology is to be designed, and (c) specifying the information technology to be designed. In parallel to these activities, (d) evaluation activities make the process cyclic. Figure 1 gives an overview of the whole design process with all activities and sources of information.

When using DUTCH it is required that the design team works in groups, because of the different types of expertise that is needed. Typically teams could consist of a variety of disciplines among which are computer scientists, psychologists, ethnographers and industrial designers.

Z.3 Groupware Task Analysis (GTA)

The design process starts by an extensive task analysis using our method GTA (Van der Veer, Lenting, and Bergevoet 1996). A task analysis of this type includes a description of the work, of the work situation and of users and other stakeholders of the system to be designed. We distinguish two task models. The first task model we make is a *descriptive* task model and is used for analysing the current task situation. The second task model is a *prescriptive* task model for the system that is to be designed. Both task models are usually done by task modelling specialists, e.g. psychologists and ethnographers. However, for the second task model all design disciplines should participate by contributing ideas which are then described by the task modelling group.

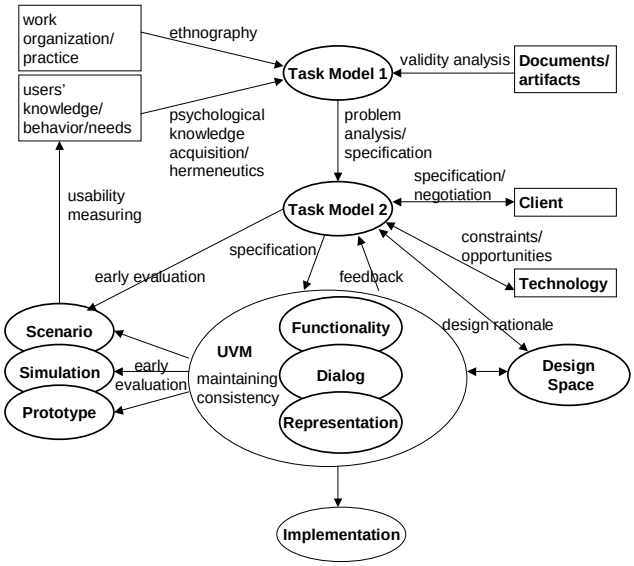

Figure 1 The DUTCH design process

Z.3.1 Analysing the current task situation (Task model 1)

In many cases the design of a new system is triggered by an existing task situation. Either the current way of performing tasks is not considered optimal, or the availability of new technology is expected to allow improvement over current methods. A systematic analysis of the current situation may help formulate design requirements, and at the same time may later on allow evaluation of the design. In all cases where a "current" version of the task situation exists, it pays off to model this. We use a combination of classical HCI techniques such as structured interviews (Sebillotte 1988) and CSCW techniques such as ethnographic studies and interaction analysis (Jordan 1996).

Z.3.2 Envisioning the future task situation (Task model 2)

The actual "step" from task model 1 to task model 2 is probably the most important one and at the same time the most open transition in DUTCH design. Based on the requests from the client of design and the problems and inconsistencies found in the current situation (task model 1), and wishes and ideas of the users and stakeholders questions for redesign are formulated. All available design disciplines can now be invited to create options for improvements or changes, without discarding each other's ideas but possibly reacting to ideas from other disciplines. After exhaustive generation of options, all disciplines should

contribute to generate criteria to evaluate all option against affordability, usability, technical feasibility, and any other type of assessment that may seem valid. Task model 2 will in general be formulated and structured in the same way as the previous model, but in this case it is not considered a descriptive model of users' knowledge, although in some cases it might be applied as a prescriptive model for the knowledge an expert user of the new technology should possess. Initially, development of task model 2 is driven by the results of the analysis of task model 1.

Z.4 Detailed Design

The detail design phase consists of the specification of the technology to be designed. This phase is started by task model 2 which contains specification on the task level but does not contain any details concerning the technology. During detailed design decisions have to be made on the look and feel of the technology including the dialogue design and hardware ergonomics/design. The task specification is used as a basis for the functionality of the technology as far as relevant to users. Using the task structures an initial grouping of functions and main navigation structure is described. In a continuous process choices are made that affect other design activities. For instance, hardware design can pose restrictions on the number of buttons or screen estate that is available. In commercial design there is often a tension between purely usability directed design and commercial issues such as product marketing.

Z.5 Designing in Teams

In DUTCH, team based design is a requirement. Teams need to work together in order to make the right decisions and to obtain a high quality end product. For example, in a typical project we found the following teams:

- Task Analysis Group
- Detailed Design Group
- Evaluation Group
- Scenario and Prototyping Group
- Management Group

We found that it is important to have one independent group that manages all the other groups. The management groups guards the time schedules, deals with delays and group conflicts and the overall project documentation. The management's main purpose is to guard the dialogue between disciplines as well as between design phases. The management group is monitoring the use of representation to communicate between design sub-disciplines, as well as between the design team and the users, stakeholders and the client.

Designing in teams can work well and potentially has several advantages. First of all, working in groups on separate issues creates a competition effect. Each group depends on each other and is being judged by other groups. If one group delivers poor work the other groups immediately remind them. On the other hand if a group has developed really new ideas, it is very disappointing if the other groups just discard their creativity or fail to see its merits. Not surprisingly, it is not unusual that certain groups get really upset by the work of others. However, this even leads to qualitative improvements of the final work as it forces the management as well as the whole team to reconsider and argue about misunderstandings and rejected proposals. Another advantage is that in this way the expertise people have is used optimally. In educational settings, the management team forms the groups and designers have to *apply* for a position. This way everyone gets to determine what he or she does, within some limits of course. In industry, a comparable "marketing" situation and resource management is in fact common practise.

Z.6 Design Representations

Each activity in the DUTCH method requires special representations. Moreover, for a single activity different representations are used depending on the purpose. Some are for internal use and other are intended for communication between disciplines or with clients and domain experts. Representations of a formal or "informal" graphical type can be created with our tool Euterpe (van Welie, van der Veer, and Eliëns 1998) while other representations are created "by hand", e.g., sketches, scenarios and role-playing.

Z.6.1 Representations for Task Modelling

Task analysis has been a field of mostly formal methods. Methods such as GOMS (Card, Moran, and Newell 1983) are rather formal and require considerable training to understand. Such representations are certainly unsuitable for communication with clients or domain experts. In our experience, at least "formal" graphical representations like task trees are reasonably understandable by client of design or domain experts as long as they are not too big. Usually parts of a tree containing 10 to 20 tasks can still be understood.

For communication within design groups the formal task models work well, especially if they are fully linked, i.e. when in fact actual representations are views on the same elements of a coherent structure embedded in a database. When task models are constructed using our tool Euterpe, the whole model can be exported to a set of linked web pages.

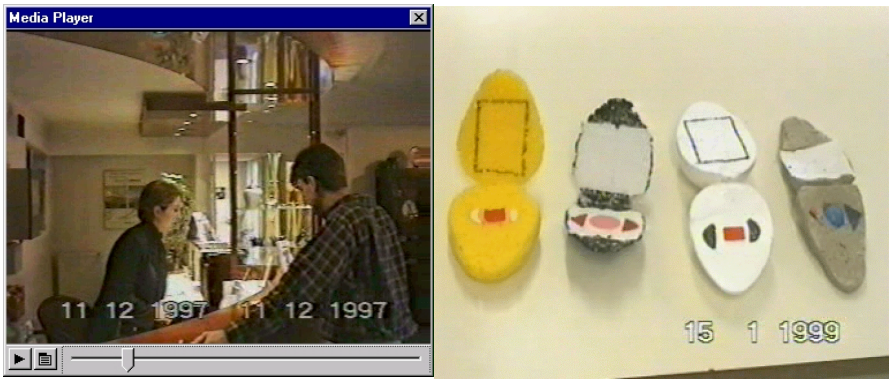

Figure 2 A video fragment and some hardware mock-ups.

Besides a linked model representation, the original material from which the model was built can also be included. We use images (e.g. scanned sketches), recorded interviews and video clips of ethnographic observations and scenario play, digitally stored and attached to the information they belong to, e.g. a certain elements of a task structure. Figure 2 shows a video fragment and some hardware prototypes for a handheld device.

Z.6.2 Representations for Detailed Design

In the detailed design phase, the design is worked out in such detail that a prototype can be made out of the specifications and finally a full-blown implementation. Representations for this phase include sketches, screenshots, interactive prototypes, hardware mock-ups and NUAN (the DUTCH version of a complete formal representation).

Especially in this phase the combination of formal and informal representations are working out well. The creative ideas are born in the informal representations and then become more detailed at each iteration. At the end of process a detailed specification is handed to the implementers and this specifications need to be unambiguous. If not, the final product may not have the intended look and behaviour.

Figure 2 shows some hardware mock-ups used to determine the basic shape and texture of a handheld device. The prototypes were made of sponge, clay and polystyrene foam. Figure 3 shows an example of a NUAN interaction specification.

Z.6.3 Scenarios

Scenario analysis is a technique that can be used during the development of task model 2 as well as for the evaluation of detail specifications, both early and

late. Scenarios are informal descriptions of tasks that are likely to occur together in a specified situation including well-described user involvement. Some scenarios can be heuristically evaluated by a group of designers, possibly together with future users. On the other hand, in some cases scenarios show their value when actually *acted out*. When scenarios are acted out, the actors are instructed to try as many variations within their task description.

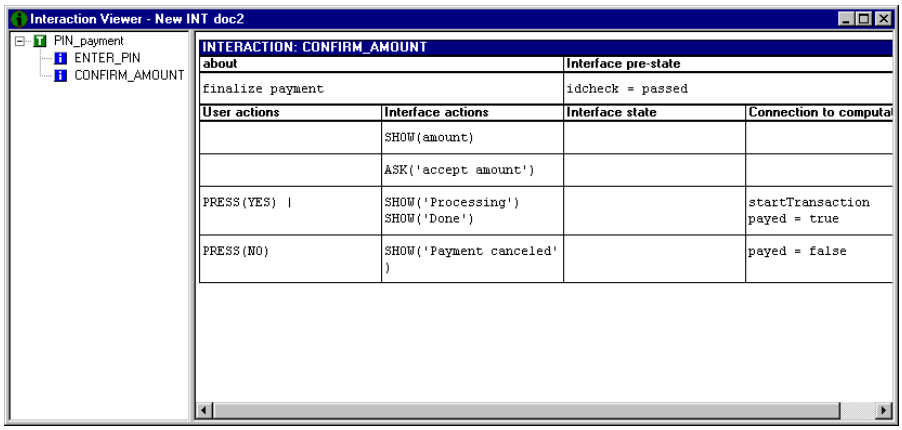

Figure 3 A NUAN interaction specification

It is useful to have some of the scenarios additionally to be acted out by intended users who in fact are sceptical or afraid of using the future technology. These actors will both show how the design could fail and how to manoeuvre around the supposed problems. To make sure nothing is overseen the whole scenario is videotaped. After several scenarios have been played a *claims analysis* is done. Together with each scenario, claims have been identified and during this analysis it is being evaluated whether or not the claims really hold.

Z.7 Formal Methods and Creativity

Formal methods allow precision and informal methods allow vagueness and creative interpretation. At some point in the design process both are needed. In our experience we found that formal and informal methods can work complementary. At some point details need to be worked out and by using formal methods hidden faults and unnoticed issues may be found.

Another issue is communication. When design groups need another group's results as input, the work needs to be 100% clear and unambiguous. Using a formal method is never a guarantee for unambiguous descriptions but it makes it easier to understand a description and finds imperfections. This is facilitated because formal method, have explicitly defined syntax and semantics.

Z.8 Does it work?

So far, DUTCH design has been spread out into regular practise in University design education in the Netherlands (4 Universities), Romania (2 Universities), as well as in Dutch post academic education (3 institutes). In Industry it has been applied in at several large design enterprises in Europe, e.g., it is regular practice in a design company in Austria for the design of safety critical systems. In all these cases, we learned that the design team claimed to benefit strongly from the facilities to combine formal representations with "informal" representations like scenario play, user evaluation of sketches, and video clips, guided by the structure of design activities and their interdependencies. We developed special design cases for education, focusing on envisioning design. The illustrations in this paper are taken from the elaboration of a student exercise to design new procedures and related technology for future regular hotel guests for hotel chains, to accommodate simple administrative procedures and to provide optimal information as well as some nice and trendy travel assistance.

Z.9 Conclusions

DUTCH is a design method that uses both formal and informal representations. We found the combination of these representations to be useful and desirable. The formal representations give precision when needed while informal representation work well to communicate ideas within design groups as well as to clients.

Z.10 References

Card, S.K., Moran, T.P. and Newell, A. (1983), *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Ass, Hillsdale.

Jordan , B. (1996), *Ethnographic Workplace Studies and CSCW*, in: D. Shapiro, M. J. Tauber and R. Traunmueller, The Design of Computer Supported Cooperative Work and Groupware Systems , North-Holland, Amsterdam.

Sebillotte, S. (1988), Hierarchical planning as a method for task-analysis: the example of office task analysis, *Behaviour and Information Technology*, vol 7(3), pp.275-293.

Van der Veer, G.C., Lenting, B.F. and Bergevoet, B.A.J. (1996), GTA: Groupware Task Analysis - Modeling Complexity, *Acta Psychologica*, no. 91, pp.297-322.

van Welie, M., van der Veer, G. C., and Eliëns, A. (1998), Euterpe - Tool support for analyzing cooperative environments, *Ninth European Conference on Cognitive Ergonomics*, Limerick, Ireland.