Design in Architecture, Software Engineering and Mechanical Engineering

A comparative study

Isabelle M.M.J. Reymen
Eindhoven University of Technology
Stan Ackermans Institute
Eindhoven
The Netherlands
isabelle@win.tue.nl

ABSTRACT

The awareness about the gap between general design theory and design practice is increasing. Design practice is not really served with the results of current design theory. To build a bridge between theory and practice, design researchers should know what is really going on in practice. To explore design practice and to find the most important characteristics of design situations, I have chosen an empirical approach based on case studies in which design projects in different disciplines are compared. In each case study, an individual designer is interviewed and the design documents are analysed. The results in this article are based on two architectural projects, two software-engineering projects and two mechanical-engineering projects. The cross-case analysis has resulted in descriptions of design situations in these disciplines. A preliminary design frame to describe design situations in different disciplines has been derived. Based on similarities and differences in the descriptions, conclusions concerning design theory, design education and design practice are given. The most important conclusions are the following. First, designers are often not aware of their design process, but focus mainly on the product. Second, software designers more often than architects and mechanical engineers use methods to structure their overall design process.

1 INTRODUCTION

Discovering the domain of design methodology, I made two observations. First, I found an increasing awareness about the gap between design theory and design practice. This gap is for example described in (Dorst 1997) and (Coyne 1995); (De Vries 1994) gives an overview of researchers who observed it in the area of architectural design. Second, I observed that ‘multi-disciplinary design’ is a trendy slogan. But what are the characteristics of design situations in the various design disciplines? Most designers are specialised in just one discipline and know little about the other. Attempts to develop general design theory, useful for multi-disciplinary design, are often based on only one design discipline.

These two observations are starting points for my research. Because design researchers should know what is really going on in practice, I have chosen to get my input especially from design practice. In this design practice, I am looking to more than one discipline. A Japanese proverb explains why: “A fish doesn’t know what is water, because it doesn’t know what is ‘no water’.” I have chosen to explore designing in the disciplines of architecture, software engineering and mechanical engineering.

The main goal of my research project is to describe design situations in different design disciplines in practice. A design situation covers all the important aspects related to designing during a design project. This description of design situations is important to give better design education, to develop better tools and to aid multi-disciplinary designing. Design literature already offers a lot of knowledge about aspects of designing. I give a non-exhaustive overview of important aspects of design situations by exploring current practice. The overview is not complete, because of the time limitations of my project and because of the diversity of projects in design practice.

My research is defined in the context of the Stan Ackermans Institute (SAI) of the Eindhoven University of Technology (EUT). This institute co-ordinates postgraduate programmes in technological design and Ph.D. design projects in many disciplines. My research proposal is titled "Towards a multi-disciplinary framework for design." The Institute has defined this project to support their design courses with design research. SAI provides a unique opportunity to work on design theory, starting from concrete design projects in many different disciplines.

My choice for case studies is explained in chapter 2. The results are described in chapter 3. In the discussion of chapter 4, conclusions based on the results of a cross-case analysis are described. Chapter 5 concludes with a summary of the results.

2 METHODOLOGY

To describe design situations in practice, the goal of my research project, I conduct an explorative study. Important aspects of design situations in different disciplines are inventoried, described, compared, and modelled. This is done by executing a series of case studies and doing a literature study. In the first section of this chapter, the general research approach is justified. In sections 2.2 and 2.3, the literature study and the case studies are described. The cross-case analysis and the verification of the results are described in section 2.4. The chapter is concluded with the methodology of the design frame.

2.1 General Research Approach

The methodology of my research project is based on methods of the social sciences. I have chosen an empirical approach to do qualitative research. The choice for an empirical method is a result of the first starting point of my research: looking at design practice. Characteristics of and criteria for qualitative research can be found in (Wester 1991) and in (Yin 1994).

The second starting point of my research is to look at more than one discipline. This has been implemented by choosing the design disciplines architecture, software engineering and mechanical engineering. Since I have limited time and I need enough empirical data per discipline, I have chosen only three disciplines. I have chosen architecture because it has already played an important role in design methodology, and it is the discipline I am most familiar with. Mechanical engineering has also offered much knowledge to design methodology. It can also be seen as a typical engineering discipline. Software engineering is a new design discipline that tries to find its methodology. Together, these three disciplines are responsible for a wide range of products and they differ enough in their design approach.

2.2 Literature Study

The literature study comprises a study of the general design literature and of some literature specific for the different design disciplines. For the specific design literature, I compared the content of the most dominating books in the different disciplines to find the most important topics concerning designing.

2.3 Case Studies

The choice for the case-study approach is motivated in the next sub-section. For the execution of the cases, a case-study protocol is followed. This protocol is described in section 2.3.2. The execution of pilot cases to test the protocol is described in section 2.3.3. The choice for the cases is given in the last sub-section.

2.3.1 *Motivation Case-Study Approach*

To explain my choice for a case-study approach, I recall two definitions for this type of inquiry from literature. The first one is found in (Yin 1994). “A case study is an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident.” The second one is a translation of the definition given by (Wester 1991). “The case-study approach is an inquiry of one or more cases going into depth by putting central the complex relations in which the case functions.” Design situations are contemporary complex phenomena with complex relations between the different aspects. Hence, my conclusion is that case studies are well suited for investigating design situations. The case-study approach, its characteristics and its types are further described in (Yin 1994), (Wester 1991) and (Verschueren et al. 1995).

I prefer the case-study approach above the widely used method of protocol studies in design research. For a description of protocol studies, see the special issue of Design Studies (Dorst 1995). Protocol studies are executed in laboratory conditions and cover often only a part of a design project. I am interested in the whole design project and not in detailed information about short time periods. Interviews are more appropriate to analyse long-term processes. The design projects for my cases are executed in their natural environment.

2.3.2 *Case-Study Protocol*

The course of one case study is described in my “case-study protocol.” This is a document that contains all the procedures and general rules that are followed to execute one case. This instrument increases reliability, continuity between cases and guides me in carrying out the case studies. The importance and design of a protocol is described in (Yin 1994).

The important aspects of the case-study protocol are now described in more detail. The data for the case studies are obtained by doing one or two interviews of one hour and a half and by an analysis of the documents of the design projects. The first interview is prepared by the designer and by myself, using a question list to inventory the important aspects of design situations. During the first interview, questions to the designers are concerned with the product and the design process. Most of the questions relate to design issues, but the link with management of the design process is also made. The designers are also asked where they see bottlenecks in the design process. The interview is tape-recorded and notes are made. The analysed documents are the documents made by the designer during the design project. The raw data are structured in a classification system, called the design frame. This is necessary to compare the different cases in a cross-case analysis. The design and purpose of this design frame is explained in section 2.5. The designers are asked to tell how they see their design situation. Because I interpret the answers on the questions and the content of the documents, I ask the designers to check my description of their design situation and to correct my view. This is done in a second interview or by mail. The results of this check are processed and finally, a summary report is made.

2.3.3 *Pilot Cases*

Two pilot case studies were executed to test the case-study protocol and the question list, and to make a time planning for the next cases. One designer from the Faculty of Mathematics and Computing science and one from the Faculty of Mechanical engineering of the EUT were interviewed with the preliminary question list, following the procedure of the case-study protocol. The interviews were evaluated on contents and procedure. The question list and protocol were improved. These pilot case studies confirm the gap between theory and practice. I started the first case with questions phrased in the terminology of the theory. I had to change the questions because the designer did not understand me.

2.3.4 *Choice of Cases*

The choice of the cases for my research is inspired by replication logic (Yin 1994). This means that every case serves a specific purpose within the overall scope of inquiry. This is called the logic of theoretical sampling. Theoretical sampling is opposed to statistical random sampling. Statistical random sampling follows sampling logic (Yin 1994); the cases are sample units. For statistical random sampling, also a higher number of cases is necessary than for theoretical sampling. Following replication logic, two types of replications can occur. For literal replication, cases deviate minimally and can predict similar results; for theoretical replication, cases deviate maximally and may produce contrasting results, but for predictable reasons. In my research, literal replication may be found within a discipline, whereas theoretical replication can be found between disciplines. Table 1 illustrates the choice of the cases.

Cases are typical examples of design projects in their discipline. These are chosen to differ from each other by experience of the designer and by the product. Almost all the projects can be called multi-disciplinary. The interviewees are individual designers, interviewed at the end of their design project. I have chosen for the design process of an individual designer, to limit the scope of the research. By choosing the end of the design project, designers can reflect more easily on the design process and project. They have a better overview of the design project. I presume that designers can easier “step out of the designerly way of thinking” (Cross 1994) at the end of the design project.

Twelve case studies are executed, four in every discipline. In every discipline, two novice and two expert designers are interviewed. A first source of cases is found in the design projects carried out at EUT. The interviewees are novice designers who have just finished their first large design project. This can be the final project to get their degree, the final project of one of the design courses of the SAI or a Ph.D. design project. The junior designers are chosen to tell me about their first design experiences and about the important aspects of designing they learned and did not learn during their education.

Professionals in industry are a second source of cases. The professionals are experts in their design discipline and have a lot of design experience. The design projects are mostly more complex and executed by more than one person. The expert designers are mostly also project leader and thus have a good overview of all the important aspects of a design project. This compensates for the fact of interviewing just one person. The purpose of the expert cases is to obtain knowledge of experienced designers working in teams. Together, the novice and expert cases cover a wide range of design projects.

The sequence of the cases is as follows. First, the two pilot cases are executed. Then, the cases with novice designers are performed. For two of these cases, the design projects of the pilot cases are chosen. The data of the pilot cases is added to the data received in these “real” cases. The cases are grouped per discipline, because these studies are combined with a literature study of designing in that discipline. After these six cases, a brief cross-case analysis is made to analyse and summarise the obtained data. At the end of this reflection period, the choice for the expert designers is made. In the next six cases, the results of the cross-case analysis are presented to the professionals to react on these.

Table 1: **Choice of Cases Following Replication Logic**

<table>
<tr><td colspan="2">THEORETICAL SAMPLING</td><td>STATISTICAL RANDOM SAMPLING</td></tr>
<tr><td colspan="2">Every case serves a specific purpose
Few cases
Replication logic</td><td rowspan="3">Cases are sample units
Many cases
Sampling logic</td></tr>
<tr><td>Literal replication</td><td>Theoretical replication</td></tr>
<tr><td>Within a discipline
Minimal deviation
Predict similar results</td><td>Between disciplines
Maximal deviation
Produce contrasting results</td></tr>
</table>

2.4 Cross-Case Analysis

After the case studies, a cross-case analysis is made. The cases within one discipline and in different disciplines are compared for similarities and differences. The highest common denominator of design situations in the different design disciplines is sought on a high level of abstraction by applying "analytic generalisation." (Yin 1994) describes this method to generalise cases as a method in which a previously developed theory is used as a template with which to compare the empirical results of the cases. This method differs from "statistical generalisation." Yin uses the term "analytic frame" for this previously developed theory. My design frame can be seen as this previously developed theory.

One of the differences between the different design disciplines is the terminology. The different disciplines share some terms. However, some have a different content or meaning. Such terms are called homonyms. Synonyms are pairs of words that have an analogue meaning. On a high level of abstraction, the comparison is based on synonyms. The vocabulary used to describe the cases is a mix of words taken from the practice and the literature of the different disciplines. To come to a uniform terminology, I introduce my own semantics of these words.

As a result of the mutual comparison of the cases, a description of design situations in every discipline is made. These descriptions are compared to make a general description of design situations. The results reflect similarities and differences between the cases and disciplines. The design frame can be situated on an abstraction level on which all the cases show similarities. It contains the common aspects of design situations in different disciplines. The description of design situations in one discipline reflects similarities between the cases in one discipline. The general description of design situations reflects similarities between the cases in different disciplines. The different descriptions of design situations in the disciplines reflect differences between these disciplines. Conclusions useful for design theory, design education and design practice are based on similarities and differences in the descriptions of design situations in different disciplines. Table 2 gives an overview of

Table 2: **Overview Results Cross-Case Analysis**

SIMILARITIES		DIFFERENCES	
In one discipline	*General*	*In one discipline*	*General*
	Design frame		
Description design situation in one discipline	General description design situations		Description design situation in one discipline
	Conclusions from comparing design situations in different disciplines		Conclusions from comparing design situations in different disciplines

the results of a cross-case analysis, classified by similarities and differences. Differences within one discipline are not analysed, because I am mainly interested in similarities and differences between disciplines.

The results of the cross case analysis are verified by again asking the opinion of the interviewees and by comparing these to other empirical studies. Finally, a synthesis of the case-study results and the literature is made to answer the research question. Figure 1 gives an overview of the research methodology.

Figure 1: **Overview Research Methodology**

PRACTICE			THEORY	
Case studies 2 Pilot cases			**Literature study**	
			General design literature	Design literature of disciplines
4 cases in Software engineering	4 cases in Architecture	4 cases in Mechanical engineering		
Cross-case analysis				
Design situations in Software engineering	Design situations in Architecture	Design situations in Mechanical engineering		
General description of design situations				
Similarities and differences between design situations				
Verify results				
Synthesis to answer research question				
Recommendations				

2.5 Design Frame

During my research project, a design frame is gradually developed following the "grounded theory approach." This is a systematic procedure for qualitative research. Theory is developed step by step, based on systematic obtained and analysed research data. This procedure is described in more detail in (Wester 1991) and (Verschueren et al. 1995).

The design frame is a structured common vocabulary to describe design situations in practice. The terminology is a mix of English words from different design disciplines. I try to choose the words close to the terminology used in practice. The links to the designers' own vocabulary has to be clear. The goal is to describe designing in three disciplines with the same terminology, in a manner that is recognised by designers in every discipline. The frame is a first attempt to build a bridge between theory and practice. It is used to structure the question list for the interviews, to transcribe the data of the cases, to classify the literature and to compare the different cases.

The frame is developed in dialogue with the cases and the literature. Starting from literature and the pilot cases, some assumptions are made about what could be general and what could be specific for the different disciplines. This results in a preliminary design frame. After the first six cases, a cross-case analysis is made and a new version of the frame is developed. The same is done after the cases with the expert designers. After the results have been verified, a new version of the design frame is made. The final version is made when the synthesis is completed. Figure 2 shows the development of the design frame.

Figure 2: **Development of Design Frame**

	THEORY		PRACTICE
	Literature study		**Case studies**
			Pilot cases
		Design frame 1	
			Novice designers (6) Cross-case analysis
		Design frame 2	
			Expert designers (6) Cross-case analysis
		Design frame 3	
			Check results
		Design frame 4	
t	Synthesis		Synthesis
		Design frame 5	

3 RESULTS CROSS-CASE ANALYSIS

Until now, the pilot cases and the first six cases in the different disciplines have been executed. Table 3 gives an overview of the cases. Section 3.1 gives a short summary of the first six cases. These cases have been compared in a cross-case analysis. A preliminary design frame has already been developed and is described in section 3.2. Descriptions of design situations in the three disciplines are made. Summaries of these descriptions are given in section 3.3. The general description of design situations is not yet presented in this article. Conclusions from comparing descriptions of design situations of the three disciplines are described in chapter 4.

Table 3: **Overview Cases**

	SOFTWARE ENGINEERING	ARCHITECTURE	MECHANICAL ENGINEERING
PILOT	Pilot case 1		Pilot case 2
NOVICE	Case 1	Case 3	Case 5
	Case 2	Case 4	Case 6
EXPERT	Case 7	Case 9	Case 11
	Case 8	Case 10	Case 12

3.1 Cases

Short summaries of the analysed design projects in software engineering, architecture and mechanical engineering are given below. The complete case-study reports contain also information about the case-study procedures, the transcribed data, the analysis of the documents and the question lists. After finishing all the case studies, the case-study reports will be published as internal report of SAI.

3.1.1 *Case 1*

“Sybar”- Development of a motion-analysis system for rehabilitation medicine at the V.U. Hospital in Amsterdam, E. Hautus

The designer followed the post-graduate programme in technological design “Software Technology” at the EUT. This resulted in a Ph.D. design project at the Faculty of Mathematics and Computing science of the EUT. The project was executed in the V.U. Hospital in Amsterdam at the Department of Clinical Physics and Engineering. The project took three years. The system is in use.

The “Sybar” project was a complex project, combining software design with hardware and mechanics. The functionality of the system was extended in a number of steps, using the rapid prototyping method. OMT, one of the methods for object-oriented design, was used to develop the system.

The analysed documentation consists of the project report of the post graduate programme and the Ph.D. thesis.

3.1.2 *Case 2*

The design of a new graphical user interface for Ex*Spect*, P. de Crom

The designer followed the post-graduate programme in technological design, “Software Technology” at the EUT. The project was executed in charge of Bakkenist Management Consultants, a Dutch consultancy office. The first part of the design project was developed during this post-graduate design programme, the second part in a co-operation between the Faculty of Mathematics and Computing science of the EUT and Bakkenist Management Consultants. The product is delivered and the project is finished.

A new graphical user interface had to be designed for the existing version of Ex*Spect.* Ex*Spect* is an Executable Specification Tool, used for system engineering to specify complex distributed systems. OMT was used as a design method. The project was very well structured and documented in the early phases. The implementation phase took more time than planned, was not very well structured and almost not documented.

The analysed documentation consists of specification documents and the final report of the post-graduate programme.

3.1.3 *Case 3*

The “marché Rose”- cultural trade centre in Bamako, J. Bierman

The design project was executed as a final project at the EUT, Faculty of Architecture, Department Production and Realisation. With this project, the designer won the Archiprix 1996. This is a competition for the best design by a Dutch student of one of the institutions that teach design in the Netherlands. The design is finished but will not be executed.

The project was a combination of research and design. The designer studied and made a structural sketch for the trade centre of Bamako, capital of Mali. A design for a market place, The Marché Rose, was made. The designer tried to incorporate a lot of different aspects and worked on different levels of abstraction. Almost all the decisions were well underpinned, referring to the information collected during the research. For this reason, the designer was called an “analytic” designer.

The analysed documentation consists of the final report of the study in architecture and some maquettes.

3.1.4 *Case 4*

Str@tegitectuur, C. Arts

The design project was executed as a final project at the EUT, Faculty of Architecture, Department Production and Realisation and Department of Architectural Design. Four sub-projects were executed. With one of this sub-projects, the designer was nominated for the Archiprix 1997. None of the projects will be executed.

For every sub-project, the designer developed constraints. These constraints form the rules of a “game”. He played the game to develop the concepts of the design. When the results didn’t please him, he changed the rules of the game until he felt happy about the result. This strategy helped him to take decisions. The different rules he started from in the sub-projects are: constraints for the playfield of a laser quest, a literal translation of the program of demands for a building for family of very ill children near a hospital, some energy rules for using transparent material, and constraints of form and structure for a student hotel.

The analysed documentation consists of the final report of the study in architecture.

3.1.5 *Case 5*

Design of a photo-voltaic/thermal hybrid panel, D.W. de Vries

The designer followed the post-graduate programme in technological design, “Computational Mechanics” at the EUT. The first part of the design project was developed during this post-graduate design programme, the second part at the Faculty of Mechanical engineering, Department Mechanical Energy Technology in co-operation with Shell Solar Energy B.V. The designer is in the last year of his Ph.D. project.

The designer had to design and built a prototype of a photo-voltaic/thermal hybrid panel, which can be placed on the roof of a house. The combi panel converts solar energy into both heat and electricity. It is a multi-disciplinary project, in which thermodynamic, optical (physics of PV-systems), mechanical, architectural, production-technical and economical problems had to be solved. No overall design method was chosen. First, concepts were generated using a list of demands. Then, a model was made for some concepts to estimate the yearly electrical and thermal field. The results were compared and a prototype was built to check the model. After doing experiments, the concepts left behind were evaluated. The resulting concept was optimised.

The analysed documentation consists of the final report of the post-graduate programme.

3.1.6 *Case 6*

Design of an axially controlled spindle unit for high precision diamond turning, M. Renkens

The designer followed the post-graduate programme in technological design, “Mechatronic design” at the EUT. The first part of the design project was developed during this post-graduate design programme, the second part at Philips Research Laboratories Eindhoven in co-operation with the Faculty of Mechanical engineering, Department Precision Engineering. The project is finished.

By applying a mechatronic approach, a new concept of a spindle unit was developed with improved performance. The total project has been split into two parts. The first part of the project was focused on a reduction of the axial error motions and the increase of the axial stiffness of a precision spindle. The second part has been focused on the development of an axial motion control of the spindle. The feasibility of the presented concept has been demonstrated on the basis of the prototype that has been built and tested. A patent application of this design and the corresponding method of making non-rotationally symmetric products have been made. A supplementary development is necessary to let the product operate in a production environment. It was a very multi-disciplinary project combining mechanics, electronics and different make-technologies.

The analysed documentation consists of the project report of the post-graduate programme and the Ph.D. thesis.

3.2 Design Frame

The preliminary design frame is described in this section. First, the relation between the design frame and design situations is described. Then, the content of the design frame is given. In chapter 2, the design frame is already situated in the research methodology and the importance of the frame is explained.

3.2.1 *Relation between the Design Frame and Design Situations*

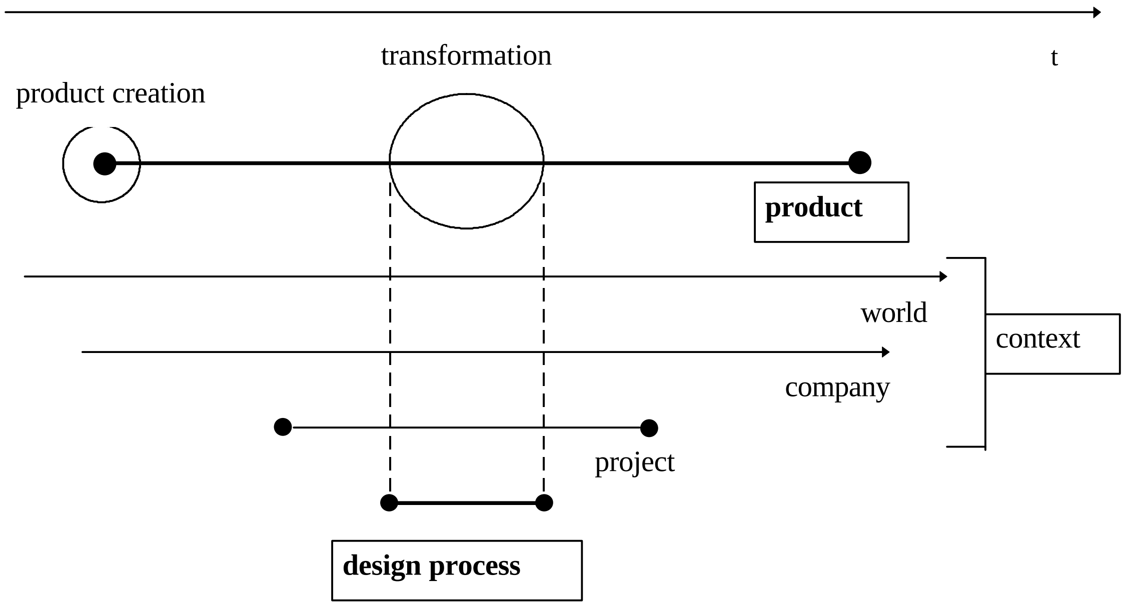
Design situations cover all the important aspects related to designing during design projects. Examples of aspects are creativity, intuition, management, the project leader, computer support, the budget, the method, product reliability, etc. All these aspects are related to each other and form a complex situation. The design frame tries to structure these aspects and provides a terminology to describe design situations. This frame is of course a simplification of reality. The structure of this frame and level of abstraction are chosen so that design situations in different disciplines can be described.

3.2.2 *Content of the Design Frame*

The most important aspects and activities of a design project are structured in the design frame. To describe design situations, many orderings of aspects are possible.
For me, the design process and the product form the basic structure of the design frame.

The design process and product can be situated against each other in time. Both can be situated in the rest of the world. A product "lives" during a certain period in time. A representation of this "life" is given in the product's lifecycle. This lifecycle starts with the creation of the product. During one or more periods in time, the product may be transformed. When this is done during a design project, these creation and transformation processes are design processes. Design projects cover also non-design processes such as marketing and management processes. The design projects are mostly executed in companies. The companies are part of the world. The project, company and the rest of the world are the context for the product and process. The design frame covers the product, the design process and their links with the context. Figure 3 situates design processes in the product's lifecycle and gives the relation with the design context.

Figure 3: **Situation of the Design Process in the Product Lifecycle and the Relation of both with the Design Context**



Important aspects of the **product** are the specifications of the system under transformation and the environment of the system. The environment can consist of people as well as machines. A product is designed to be used in its environment. A specification describes the constraints to start from and the desired end-condition of the system, including the interaction with the environment. The design is one of the possible solutions for the desired transformation.

The most important aspects of **design processes** are the activities, the people and the resources. The activities are all the actions of the designers and other people involved in a project. The most important activity during a design project is designing. That is creating the specifications of the desired transformations and executing these transformations. Management is an activity very related to the design process. Managers control the design process. Under people, aspects like social processes and organisation are classified. Resources of the design process are methods, tools, money, capacity, space, and also knowledge and skills of the designers.

All these aspects reflect three different views important in design projects. The first one is the user view. This is the view of someone who is part of the environment and who is (only) interested in the external view of the product. He/she sees the system as a black box. The technical view concerns the technical aspects of the product and the design process. The organisational view looks from the viewpoint of managers, being interested in controlling the design process. Figure 4 gives an overview of these viewpoints.

Figure 4: **Overview of Important Viewpoints in a Design Project**

Environment	PRODUCT	DESIGN PROCESS	Management
	Technical view		
User view		*Organisational view*	

Figure 5: **The Preliminary Design Frame**

PRODUCT	DESIGN PROCESS
Specifications	Activities
	People
	Resources
Relation with Environment	Relation with Management

With the design frame, **design situations** can be described. A design situation can be described as the state of the design process and the product at a certain moment. This description can be given from a certain point of view. To describe design situations, different levels and dimensions must be used. The importance of levels and dimensions is described in (Bax 1989). The next versions of the design frame will include levels and dimensions and a more detailed description and definition of important aspects. The preliminary design frame is represented in figure 5.

3.3 Description of Design Situations

Descriptions of design situations in architecture, software engineering and mechanical engineering are summarised and given below. Only the most characteristic aspects of design situations in those disciplines are described. These descriptions are based on the cross-case analysis and on literature study. The design frame is used to structure the text in a common vocabulary. The description is given from a technical viewpoint. These descriptions contain also some common value, conventions and tacit knowledge of the different disciplines. This knowledge is necessary for comparison.

3.3.1 *Description of Design Situations in Architecture*

Architecture can be described as being concerned with transforming the building environment. The *product*s of architecture are buildings in general. These are material products that create immaterial space. The *environment* of the product can be the inner environment of the building with its users, and the outer environment of the space around the building (neighbours, village, city, infrastructure). The product has a direct influence on the public space and is therefore part of the public debate. Important phases in the lifecycle of a building and its environment can be the creation process, use, transformation process, use, demolition, and recycle process. The creation and transformation processes include development, production, transport, and realisation of the building. Standard *specifications* for architects are the program of demands (functional specification, budget, surfaces, when ready to use), the situation (location) and the legislative restrictions (local, national or international). The last ones are fixed, the others can be made and changed by mutual agreement of the partners.

In the *design process* of architects, different levels of increasing detail can be distinguished. These get the following names: analysis of location and of program of demands, formulating demands, sketch design, preliminary design, definitive design (or realisation design, materialisation), and “as built”. The names of the different levels also correspond with the most important *activities* and the names of the drawings made. These periods of working on a certain level of detail are also called “phases”. The design process has a very strong milestone, namely the receiving of the building permission. The important *people* in the design process of a building can be the architect, commissioners, designers in other disciplines, executors and intermediates (project management offices). The *resources* that can be used by architects are computer aided design (CAD), drawings and sketches on paper, maquettes, calculation tools, methods for time and cost estimation, and planning tools. A difference is made between means for studying the development of the product and means for presenting to the commissioner and/or the users. Architects learn designing already during their education by executing many projects of increasing complexity.

3.3.2 *Description of Design Situations in Software Engineering*

Software engineering can be described as being concerned with transforming software. The software *product* can be embedded software as well as application software. Software is an immaterial product, basically only working with signals. In software engineering, the *environment* of the system can be people as well as machines. They “use” the software. The system interacts with its environment through the interface. Important phases in the lifecycle of the product can be the creation process, use, transformation process, use, and re-use process. The following *specification* documents can be made: domain analysis, problem statement, requirement definition, requirement specification and models of the relation of the system with the environment. Also prototypes can be made to facilitate the discussions with the user and to check the specifications.

The *design process* consists of the following important *activities*: defining the specifications, implementing the text into code, and verifying the solutions. Making specifications is an explicitly mentioned activity of software designers. The *resources* software designers must choose are a tool-set for specification, programming, verification and validation. A tool-set can comprise a design method, a development platform, compilers, Computer-Aided-Software-Engineering (CASE) tools and/or specific libraries. A language corresponds with every tool. Software engineers talk about a “specification language” and a “programming language”. For software engineering, a large number of methods exist. The “design” methods cover only a part of the design process. The methods are often not strictly followed but are used to structure the design process. All the software designers have knowledge of at least one design method and programming language. One of the most used design methods is object-oriented programming, see (Booch 1994). OMT (Object Modelling Technique), is a well known and much used technique in software engineering. An important method to quickly develop prototypes is the “Rapid Prototyping” method (Rumbaugh 1991).

3.3.3 *Description of design situations in mechanical engineering*
Mechanical engineering can be described as being concerned with transforming consumer products or production systems. This *product* can be a single product or a product for mass production. The *environment* of the system can be the consumer or operator and hardware or production-systems. Important phases in the lifecycle of the product can be the creation process, use, re-engineering (transformation), use and recycling. The important aspects linked to these phases are described in literature as “Design for X”, where X means function, safety, assembly, transport, stress, environment, reliability, maintenance, production, etc. For the *specifications* of the product, mechanical engineers mostly refer to existing products or components. They want to optimise these products by combining different components or by using different techniques.

The *design process* consists of the following important *activities*: generating concepts, modelling the system and the relation with the environment, making drawings of the system in two and three dimensions, building prototypes and making simulations to check the model by doing experiments, evaluating the results and optimising the resulting concept. *Resources* for mechanical engineers are pen and paper, Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) tools, and calculation tools.

4 DISCUSSION

Conclusions based on the results of the cross-case analysis and literature study are described in this chapter. These are preliminary conclusions. The conclusions concern design theory, design education and design practice. First, conclusions based on similarities and differences in the descriptions of design situations are described. Then, an overview of the most important characteristics of design situations in software engineering, architecture and mechanical engineering is given. In this overview, the conclusions are situated.

4.1 Similarities

4.1.1 *Designers are often not aware of their design process and focus mainly on the product. Describing their design process is seen as difficult.*
This first conclusion is recognised in the case studies and literature, both concerning the practice of designing. The designers interviewed did almost not reflect on their design process before they were asked to do so. They had difficulties with describing their design process for their thesis and for my interview. For giving these descriptions, they were “forced” to look back and oversee the process. The designers had almost no

structure and vocabulary for their design process description. Talking about the product (content) was no problem.

This conclusion is very much related to what is written in (Dorst 1997). One of his points in the image of the industrial design practice was the observation that when you are designing, you are inside a design process (thrown into a situation), and not always in the position to consider it critically and rationally. “Therefore, if you want to be in control of your design process, you must step out of the “designerly way of thinking” (Cross 1994), every now and then.” (Dorst 1997) This “stepping out” seems to be difficult.

Starting from these conclusions, the following questions could be asked. Is it necessary to bring more reflection into the design processes? If so, can we help designers during these reflection periods?

4.1.2 *The underpinning of design decisions and design rationale is underexposed in the three disciplines, in practice and education.*

A rich literature about design decisions exists in the different disciplines. However, the education and practice does not use much of this knowledge. I compared the content of books used in the design courses of software engineering, mechanical engineering and architecture at the EUT. In none of these books, explicit attention was given to design decisions and design rationale. In the case studies, the designers mentioned design decisions and design rationale as important aspects.

4.1.3 *There seems to be a basic core of design skills shared by all disciplines.*

Summarising the skills mentioned in the cases, the most important skills for designers are: knowing to analyse problems, being creative, daring to reject solution paths, and having good communication skills.

Skills linked to the first one are identification with the users, seeing the difference between primary and secondary things, working structured and keeping an overview of the product and process. To be creative, designers must be interested in and have some knowledge of a broad field of aspects. This also implies that they must see where interesting things can be found. Designers must be able to change the direction for finding a solution. Therefore, they may not stop too early and must stay critical. Communication skills are necessary to reproduce ideas for themselves and for other people. These are other designers as well as the commissioners (who speak another “language”). Designers need the means to communicate with them, for example by language, drawings or sketches, or by simulations. Good communication skills also mean listening to other people and being open for solutions of other people. Without good communication skills, it will not be possible to convince other people of the ideas of the designers, which can be necessary to “sell” the design.

4.2 Differences

Three conclusions based on differences between the cases and on literature are described in this section. The first two conclusions concern underexposed aspects in the disciplines. A difference is made between the literature and practice in the disciplines. In the literature of software engineering and mechanical engineering, non-technical aspects like creativity, communication skills and social processes are underexposed. In the practice of architecture and mechanical engineering, the rational approach using methods is underexposed. Table 4 gives an overview of the differences in underexposed aspects in the disciplines.

Table 4: **Differences in Underexposed Aspects in the Disciplines**

	NON-TECHNICAL ASPECTS		USE OF METHODS	
	Literature	Practice	Literature	Practice
SOFTWARE ENGINEERING	underexposed	awareness	awareness	awareness
ARCHITECTURE	awareness	awareness	awareness	underexposed
MECHANICAL ENGINEERING	awareness	awareness	awareness	underexposed

4.2.1 *Software designers more often than architects and mechanical engineers use methods to structure their overall design process.*
The literature on software engineering contains many hundreds of methods. The literature on architectural design contains the important work of (Jones 1980), (Alexander 1977), (Cross 1994) and many others. Literature about methods in mechanical engineering is for example (VDI 1987) and (Pahl et al. 1988). Only in the cases of software engineering, explicit use of methods was made. These methods structured the design process. Mechanical engineers and architects developed a structure for their process without using a method as a starting point.

4.2.2 *In the literature on software engineering, non-technical aspects are underexposed.*
For software engineering, I compared the content of the following books: (Sommerville 1992), (Pressman 1992), (Macro 1990), (Sodhi 1991), (Avison et al.1988), (van Vliet 1993), (von Mayrhauser 1990), (Bell et al. 1992), (Jalote 1991), and (Winograd 1996). These include the books used in the design courses software engineering at the university. Other important books can be added, but these give already a tendency. These are all recently published and a lot of them reprinted many times. Only the book of Winograd pays attention to aspects as creativity, context, and philosophy. In this book, many references to architecture are given. Of all the other books, only a small chapter of human factors in (Sommerville 1992) and a chapter about software psychology in (van Vliet 1993) is given. In the case studies, the non-technical aspects are recognised in all disciplines. Software engineers mentioned the underexposed aspects as important topics learned during designing.

4.2.3 *Architects learn designing already during their education by executing design projects.*

All young designers learned a lot of their first large design project. Architects do this during their first phase program by executing a lot of design projects of increasing complexity. Afterwards, in practice, only more constraints are defined and more parties are involved in the projects. Software engineers learn designing by doing, during second phase programs, during a Ph.D. design project, or in practice. In the case studies, none of the software engineers and mechanical engineers said to have learned designing yet. It seems that this situation was specific for the EUT. For the architects, the answer on the question if and where they learned designing was evident: during their education!

4.3 Overview

An overview of the most important characteristics of design situations in software engineering, architecture and mechanical engineering is given in table 5. This overview is structured according to the design frame. The most important characteristics of design situations, described in section 3.3, are summarised and the conclusions of the previous two sections are situated. Starting from this overview, much more differences between design situations in the different disciplines can be recognised. The blanks still have to be filled out.

5 CONCLUSION

Design situations in architecture, software engineering and mechanical engineering have been compared in an attempt to bridge the gap between design theory and practice. To start from practice, an empirical approach based on case studies was chosen. This has resulted in a preliminary version of a design frame. Similarities and differences in design situations in both disciplines have been described.

For further research, another six cases of professional designers remain. The design frame will be refined and the blanks in table 5 will be filled in. The similarities and differences between the cases will be explained.

My results can be used by design researchers and designers. Insight in design situations in practice can help researchers to take into account all important aspects of design projects. In this way, they can develop better design methods, tools and design curricula. Insight into concrete design situations of different disciplines allows the generalisation of the observations and the development of concepts for multi-disciplinary design.

Table 5: **Overview Conclusions about Similarities and Differences in Design Situations of Software engineering, Architecture and Mechanical Engineering.**

	SOFTWARE ENGINEERING	ARCHITECTURE	MECHANICAL ENGINEERING
Product	Immaterial product	Material product	Single or mass product
Specifications			Design for X
Environment			
Process	No awareness of design process		
Activities			
Resources	Tool set: many methods, different "languages"		
	Basic core of design skills		
People			
Managing			
Underexposed Aspects	Underpinning of design decisions		
	Non-technical aspects	Use of methods	Use of methods
Education		Design projects during education	

6 ACKNOWLEDGEMENT

I want to thank the following people for their comment on previous versions of this paper and for the stimulating discussions: Johan van Aken, Twan Basten, Thijs Bax, Maarten Bodlaender, Paul van Gorp, Dieter Hammer, and Peter Kroes. I also want to thank the designers in the cases for their co-operation: Chris Arts, Janneke Bierman, Paul de Crom, Edwin Hautus, Marcel Renkens, and Douwe de Vries.

7 REFERENCES

Alexander, C., Ishikawa, S. and M. Silvester et al. (1977) *A pattern language: towns, buildings and construction*. Oxford University Press, New York.

Avison, D. and G. Fitzgerald (1988) *Information systems development: methodologies, techniques, tools*. Blackwell scientific publications, London.

Bax, M. (1989) Structuring architectural design processes, *Open House International* Vol.14 No.3, pp. 20-27.

Bell, D., Morrey, I. and J. Pugh (1992) *Software engineering: a programming approach* (2nd ed.). Prentice-Hall, London.

Booch, G. (1994) *Object-oriented analysis and design with applications*. Benjamin

Cummings, Amsterdam.
Coyne, R. (1995) *Designing information technology in the postmodern age*. MIT Press, London.
Cross, N. (1994) *Engineering design methods: strategies for product design* (2nd ed.). Wiley, Chichester.
Dorst, C. (1995) Analysing design activity: new directions in protocol analysis, *Design Studies* Vol.16 No.2, pp. 139-142.
Dorst, C. (1997) Describing design: A comparison of paradigms, Ph.D. Thesis, Vormgeving Rotterdam, Rotterdam.
Jalote, P. (1991) *An integrated approach to software engineering*. Springer-Verlag, London.
Jones, J. (1980) *Design methods: seeds of human futures*. Wiley Interscience, London.
Macro, A. (1990) *Software engineering: concepts and management*. Prentice-Hall, London.
Mayrhauser von, A. (1990) *Software engineering: methods and management*. Academic Press, London.
Pahl, G. and W.Beitz, ed. by K. Wallace (1988) *Engineering design: a systematic approach*. The Design Council, London.
Pressman, R. (1992) *Software engineering: a practitioners approach*. Mc Graw-Hill, London.
Rumbaugh, J., Blaha, M., Premerlani, W. et al. (1991) *Object-oriented modelling and design*. Prentice-Hall, London.
Sodhi, J. (1991) *Software engineering: methods, management and CASE tools*. Mc Graw-Hill, London.
Sommerville, I. (1992) *Software engineering* (4th ed.). Addison Wesley, Amsterdam.
Verein Deutscher Ingenieure (1987) *VDI guidelines 2221: systematic approach to the design of technical systems and products*. VDI-Verlag, Dusseldorf.
Verschueren, P. and H. Doorewaard (1995) *Het ontwerpen van een onderzoek*. Lemma, Utrecht. (in Dutch)
Vliet van, H. (1993) *Software engineering: principles and practice*. J.Wiley and Sons, New York.
Vries de, E. (1994) Structuring information for design problem solving, Ph.D. Thesis, Faculty of Philosophy and Social Sciences, Eindhoven University of Technology, Eindhoven.
Wester, F. (1991) *Strategieën voor kwalitatief onderzoek*. Coutinho, Muiderberg. (in Dutch)
Winograd, T., Bennet, J., de Young, L. et al. (eds.) (1996) *Bringing design to software*. Addison Wesley, Amsterdam.
Yin, R. (1994) *Case study research: design and methods* (2nd ed.). Sage Publications, London.