Reihe Informationsmanagement im Engineering Karlsruhe

Michaël Prieur

Functional Elements and Engineering Template-based Product Development Process

Application for the Support of Stamping Tool Design

Band 1 – 2006

universitätsverlag karlsruhe

Michaël Prieur

Functional Elements and Engineering Template-based Product Development Process

Application for the Support of Stamping Tool Design

Reihe Informationsmanagement im Engineering Karlsruhe
Band 1 – 2006

Herausgeber
Universität Karlsruhe (TH)
Institut für Informationsmanagement im Ingenieurwesen (IMI)
o. Prof. Dr. Dr.-Ing. Jivka Ovtcharova

Functional Elements and Engineering Template-based Product Development Process

Application for the Support of Stamping Tool Design

von
Michaël Prieur

universitätsverlag karlsruhe

Dissertation, Universität Karlsruhe (TH), Fakultät für Maschinenbau, 2006

Impressum

Universitätsverlag Karlsruhe
c/o Universitätsbibliothek
Straße am Forum 2
D-76131 Karlsruhe
www.uvka.de

Universitätsverlag Karlsruhe 2006
Print on Demand

ISSN 1860-5990
ISBN 3-86644-033-2

To my parents

Preface of the Publisher

The strong competitive situation in the automotive industry pressures the companies to invest great efforts to improve not only their products but also their processes. The stamping tool development process is one of the key processes where progress could be made, particularly in the direction of efficiency, standardization and flexibility. Furthermore, IT systems supporting the processes are evolving and bring interesting improvement potentials that should be put into practice. In this domain, the main objective is to reap benefit from the systems' possibilities and to derive methodologies adapted to the applications, i.e., to fill the gaps between system evolution and process requirements.

This thesis introduces a new knowledge-based modeling method. It targets the development of functional elements that are directly associated with concrete application domains. Functional elements have been developed in order to adapt the modeling methodologies to the designers' way of thinking and guarantee the information flow along the process chain. They are central engineering objects regrouping information about the designers' know-how, processes, and standards and represent the interface between designers and CAx systems.

In the same way as features so far, functional elements are information cores; yet in this case it has to do not only with basis elements whose sum corresponds to a product model but directly with reusable elements on the product level. In this way, the idea of standardization - which is, in any case, implicitly present in design processes – has been used as the basis for modeling methods, leading to the development of functional elements of different natures. The first kinds are engineering templates, the second are integrated components, and the last category comprises functional features.

The influence of functional elements on the development processes has been evaluated. As these make it possible to provide information for the designers before the previous tasks are fully accomplished, process tasks can be parallelized and, thanks to the elements' associativity and the use of a PDM system, the product design cycle is shortened.

Further, this thesis describes the implementation of the functional element method based on the example of the stamping tool design process mentioned above. The complete process chain from the car-body component model up to the stamping tool model has been described with engineering templates.

Prof. Dr. Dr.-Ing. Jivka Ovtcharova

Preface

This thesis is the result of the work I carried out from September 2002 to August 2005 as a post-graduate student in the Research and Technology Center of the DaimlerChrysler Group, in Ulm, Germany. During these three years I worked in the Department of Product and Production Modeling in the IT for "Engineering and Processes Lab. Through several projects carried out in parallel to my Ph.D. research, I had the chance to acquire valuable experience in the automobile industry and to become very familiar with its processes and specificities. Hence, the subject of this thesis arises directly from current problems in the automotive industry, and the solutions proposed are strongly application oriented.

I would first like to express my gratitude to my doctoral thesis supervisor, Prof. Dr. Dr.-Ing. Jivka Ovtcharova from the IMI (Institute for Information Management in Engineering) Chair of the Mechanical Engineering Faculty at the University of Karlsruhe and especially for the interest she showed to my work. She left me the latitude to define the topic and the framework of my research, which is a great opportunity for a post-graduate student working in the industry: it is particularly difficult to find a common direction for the projects and the dissertation. Additionally, she provided me with helpful advice and support, being an essential factor for the success of this work. She also opened contacts to her assistants, whom I wish to thank for the informative discussions we had. I am grateful to Prof. Dr.-Ing. Martin Eigner from the University of Kaiserlautern for being a member of the evaluation committee.

My thanks also go to both senior managers who directed the department in which I worked: first Dr.-Ing. Siegmar Haasis, who took me on as an intern in his department and gave me the possibility to start a dissertation, and Robert Winterstein, who continued to place his confidence in me. I sincerely thank my associates who supported me, ensuring progress in the actions I took.

The center of interest of this work is the stamping tool development process. In other words, the dissertation proposes a solution for improving the process, from the car body component design to the stamping tool design. It could not have been done without the help of the designers in the Body-in-White Engineering Department at the Mercedes Car Group in Sindelfingen, who shared their know-how and taught me the interesting aspects of stamping tool technology and design methods. That is the reason why I would like to sincerely thank all the designers I met in this department.

Finally I am grateful to Nick Suyam, who provided me with meaningful advice for the improvement of my thesis, and Larissa Glaser for the effective corrections she made.

Michaël Prieur

Contents

Chapter 1
Introduction

1.1 Motivation

1.1.1 Challenges in the Automotive Industry

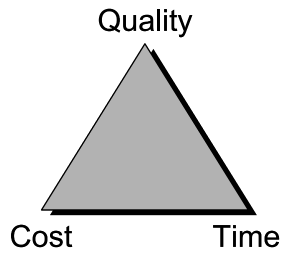


Figure 1-1: Quality, cost, time triangle.

The industry is confronted by a constant pressure resulting from three opposing factors, namely *quality, time, and cost* (Figure 1-1). This is particularly true in the automotive industry, where competition is permanently increasing and where differences between the companies, in terms of quality and technological advance, are diminishing.

Quality
Improving quality is at the top of the companies' priorities. The customer's demands are growing stronger in the direction of safety, environment, design, efficiency, and performance. Added to these issues is the problem of conformity to national legislation. The product quality is often defined as the concept of making products fit for a purpose with the fewest defects. Yet quality arises first from the product development phase where it is seen as the conformance to specifications or expectations. As there are many different definitions of the term quality [Sche-03], we will define it for this thesis as the ability of a unit to fulfill its inherent requirements. The notion of quality is relevant not only for the end product but also for each step of the process leading to it; in particular, the development process should be afforded scrupulous attention.

Time
To stay competitive in a changing market, automotive companies need to continuously reduce the product time-to-market by increasing operational efficiency. This notion has become essential for satisfying the customer's desires. Thus one key goal of the automotive industry is to be more and more efficient in order to deliver innovative and high-quality products as soon as possible. Accordingly, the time to market is one of the most important factors when estimating return on investment (ROI) for new product developments. While, for two decades, the standard development time in the automobile industry was 42 months, carmakers today envisage that this can be cut down to 18 months. This dramatic reduction in the

development cycle could not be achieved with the traditional method of design and tests. *Only new design methods and tools could make it possible to reach this goal [Stoc-03]*. As a result, most of the forward-looking organizations are beginning to move towards the Virtual Product Development Environment (VPD).

Cost
Apart from the enhancement of the product quality and performance, consumers are taking a sharp look at prices: the amounts they are willing to pay are declining. Thus, to meet customer needs and to still remain profitable, *companies are pushed to solve the paradox represented by the enhancement of product quality and the reduction of costs*. Cost cutting is mainly seen as an improvement in productivity and the elimination of waste. Companies that focus on cost-optimized designs are reaping the resulting benefit of higher product quality so that they are now forced to simultaneously the same attention on both the design and manufacturing processes.

The only way to cut costs is to be innovative in the fields of material and production costs. Most of the time, these efforts are largely supported by suppliers. Thus, for original equipment manufacturers (OEMs), benefit could be reaped by optimizing production processes but also by mandating suppliers to reduce their costs and by integrating them more effectively in the supply chain.

Complexity
Complexity could be added to the general tension triangle of cost, quality, and time. This issue is present in all the domains of product development, production, the supply chain, etc. In the last few years, major changes have occurred in the automotive industry, which has seen a dramatic reorientation. *Indeed production has switched from a mass-production strategy, where products are sensibly identical to a made-to-order principle characterized by a high level of customization*. To meet the customer's individual desires, differentiation has become a decisive criterion. This strategy has led to the creation of micro-segments, which, in turn, has increased the number of car models and variants [Lieb-03]. According to M. Hara [Hara-04], automotive companies cannot target sales volumes of hundreds of thousands units of a single type of vehicle and, as a consequence, flexibility and efficiency are required. Independently of the number of variants, the complexity of each product is increasing because of the introduction of new technologies, particularly electronic systems [Beck-03].

1.1.2 Evolution of Information Technologies

Definition 1-1: Information technology (adapted from [Kuma-03])
Information technology (IT) encompasses all forms of technologies used to create, store, and exchange information in its various forms.

The evolution of information technologies affects all the aspects of product development. It not only involves modifications of the tasks and the communication but also necessitates reorganizations and adaptations of the processes. Thanks to information technologies, some application domains such as the traditional development field have been accelerated and others such as virtual reality have been created and promoted.

In the engineering field, information technologies are mostly represented by CAx systems and product data management (PDM) systems. Computer-aided design

(CAD) found its first application in the graphical representation of drawings on monitors. Since their first appearance, CAD systems have evolved considerably and are now indispensable tools for product development. Not only do they support the three dimensional representation of product geometry they also enable engineers to formalize their knowledge and integrate it in the product representation. In fact, CAD systems offer many possibilities in the direction of parameterization, user-feature creation, rules, and script programming as well as model or information linking.

In addition to CAD systems, there are a number of tools that complete the spectrum of possibilities offered to designers, above all computer-aided engineering (CAE), computer-aided manufacturing (CAM), computer-aided innovation (CAI), and computer-aided process planning (CAPP). PDM systems integrate and manage the processes, applications, and information that define the products across various systems and media and across all the phases of the product lifecycle.

1.1.3 Current Situation in Car-Body and Stamping Tool Development

As stated, information technologies are evolving and open more and more possibilities for the development process. However there is a significant delay between the introduction of new IT systems or new functionalities and their practical implementation in the development process. Therefore, even *if new IT systems are introduced in the development process, all the advantages they offer are not fully exploited*. In particular, four deficits can be identified:

- **Little exploitation of the functionalities intended for knowledge integration**: CAD systems offer significant advantages for the parameterization of models; in particular, they allow quick and associative modifications of designs. However the possibilities provided by these systems to integrate knowledge in the model are still underexploited by designers. There are two reasons for this situation: first designers do not have the necessary qualifications and experience to handle knowledge integration. Second, knowledge insertion requires time investment and preliminary work that is not compatible with the designers' daily tasks.

- **Few design methods and little designer support:** There are directives to standardize the models and to specify how the design results should be presented, but the method to come to these results is not defined. In fact, designers do not have enough support for the modeling task at their disposal and employ methodologies that are still geometry oriented. In addition, they lack sufficient coaching and training [Lind-02].

- **Disconnection between the steps of the process chain:** It is not possible to reconsider and reorganize the processes every time software opens new possibilities. Processes hence evolve slower than software. Consequently, processes do not profit from the opportunity to link the steps with each other efficiently: CAx systems and PDM systems make it possible to establish connections between models or elements of a model, to update them in the case of design modification; they also enable information exchange between the various steps of the process chain.

- **Weak consideration of simulation in the design stage**: The employment of simulation of any kind has increased in the last past years, but it is seen as a means to validate the design and not to accompany and support the product definition process. As the performance of such systems is expanding significantly, it is to be expected that they will gain more and more importance in the development process. For this reason, the place of the simulation in the process has to be reconsidered and the connections between the simulation stage and the other steps should be redefined.

1.1.4 Origin of the Topic

This thesis was written in an industrial environment. Therefore its topic results from observations of real needs and expectations in the industry. The first set of such requirements was directly given by the business units that commissioned the research department with the projects. The second set of needs was identified during the run of these projects.

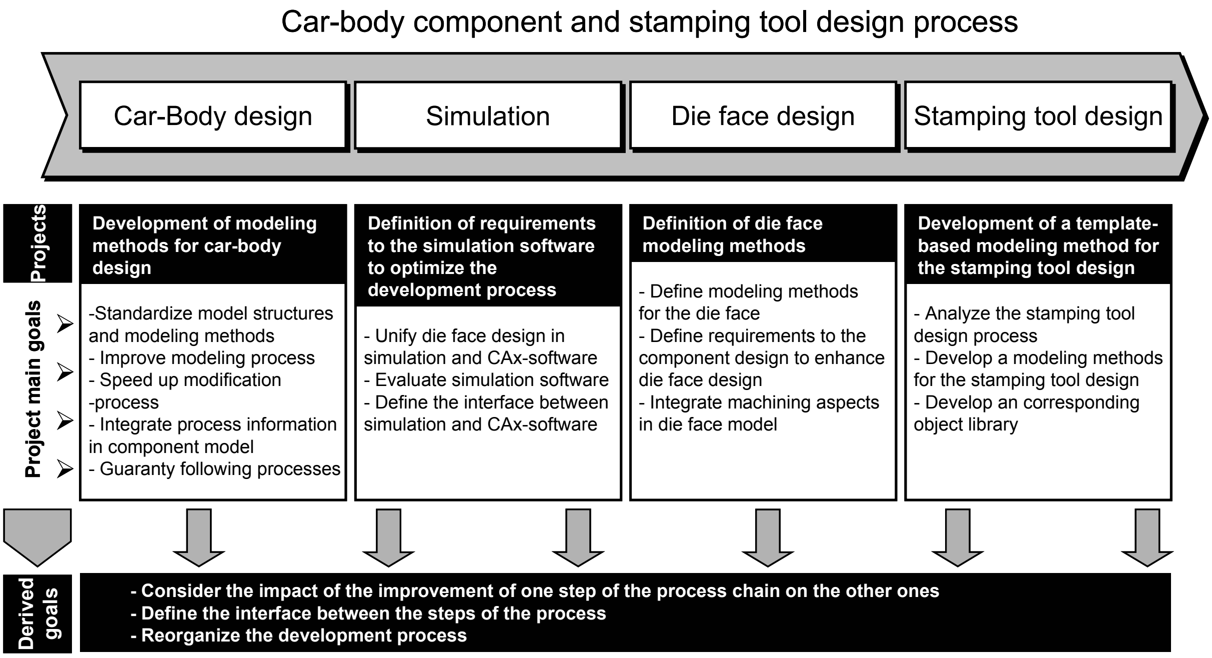


Figure 1-2: Car-body component and stamping tool design process and related research projects.

Figure 1-1 indicates the simplified process chain starting from the car body component design up to the stamping tool design across the simulation and the die face design. The research projects whose topics are in line with this process chain have been represented under each step together with their main goals. The contents of the projects show that the each one focused on relatively narrow domains and that there was not one project that covered the overall process chain. For the improvement of each step, only aspects of adjacent steps were considered. However, during the work for these projects, it became clear that *the interconnections between the steps make it is necessary to consider all of them at the same time and, in this way, further requirements for this thesis could be identified.*

According to [Zura-03], much effort is invested in the domain of electronic and software development for the automobile industry. Thus it is logical to ask whether it is necessary to carry out research in the field of mechanical product development and, in particular, in stamping tool development, which is an established domain . W. Dürheimer [Dürh-02] offers one answer to this question by demonstrating that mechanical systems are the basis of automobiles whose concept has not undergone radical modifications in the last hundred years. Second, the topic of this thesis is not to define new technical solutions for the stamping tool but rather to provide solutions to support the design processes. Thus, it is not a contradiction to develop new design methodologies for products that are considered as standard.

To conclude, the title given to this thesis is "Functional Elements and Engineering Template-based Product Development Process – Application for the Support of Stamping Tool Design". The beginning of this title indicates that this thesis will, first of all, provide a general solution for the product development using so-called functional elements, and the second part is a reference to their application in stamping tool design.

1.2 Objectives of the Thesis and Expected Benefit

1.2.1 Objectives

The common, admitted answer to the issues mentioned in the previous section is innovation: according to [Pete-98], innovation is the motor driving the activities aimed at resolving the problem of production cost, development time, and product quality.

Definition 1-2: Innovation (adapted from [Barn-53])
An innovation is any thought, behavior, or thing that is new as it is qualitatively different from existing forms.

To reach the goal of competitiveness, innovation has to be assured not only for the final product but also for the process leading to it. Indeed, whereas the general public thinks of innovation as a product, companies understand it as a process [Sage-00]. Thus design processes are not static and have to be improved more and more. The need for progress is so much the greater that new design technologies and systems are regularly introduced. Their complexity however means that they cannot be introduced without defining the way to work with them. Thus, new design methods are needed when new tools are released.

Objective 1: Provide an overview of the current product development methodologies and technologies
The first goal of this thesis is to portray the situation in the product development domain: the attention will be centered on the design and modeling methodologies currently available. Since features have been recognized as key elements to support product modeling and process planning, this work will first give an overview of feature technology, its various definitions, and its application domains.

The second focus of attention will be the description of the available solutions supporting the design process chain in general, namely the use of knowledge, the integration of machining information in design, or the place of simulation. Further, the issue of modeling methods has to be examined with an emphasis placed on model parameterization and reusability. Finally, the topic of concurrent engineering should be approached since it represents a general aim in the product design and manufacturing process.

Objective 2: Record the current deployed stamping tool development process including component design
As there is a particular demand to improve the stamping tool development process, this has to be picked up and represented to clarify its current state. Yet this process should not be considered as isolated. It is hence necessary to record the adjacent activities as well. Consequently the design of the car-body component that delivers the inputs of the stamping tool development will also be described, as well as the production of the stamping tool receiving its outputs. In addressing the process mentioned, the priorities will be activity sequencing and the information flow.

Objective 3: Identify the weak points of the currently deployed process to define the requirements for an improved process
After the recording of the process, its weak points have to be identified in order to define the main goals of the thesis. In this context, identifying weaknesses includes evaluating the necessity of each task of the process chain, determining the redundancies in the process, assessing the efficiency of each step of the process chain, detecting the gaps in the information flow, evaluating the validity of the data, and evaluating the interface between applications. To rate these criteria, it is necessary to consider the CAx process and PLM as references as well as the possibilities provided by state-of-the-art software.

Objective 4: Provide design methodologies for each step of the stamping tool design process
The improvement of the process chain in car-body and stamping tool development will be based first on the improvement of each step of the process. In fact, it is essential with respect to user bias to provide solutions in which users can find perfection for their own work. Therefore, for each step of the process, adapted design methods have to be defined.

Objective 5: Enable associative links and information flow between the steps of the process and arrange them
In addition to the improvement of each step, the overall process chain and the interactions between the steps have to be considered. According to the result of the process analysis, the entire process working and sequencing will be adapted so as to be perfected. The criteria to consider here are efficiency (quality of the delivered data and timely optimization) and flexibility.

Objective 6: Use simulation to enhance car-body component quality and to improve the design process
Due to the increasing importance of simulation in the development process, the evaluation of its influence on the process will be a center of attention. First, its place in the process has to be evaluated and, if necessary, adapted; second, its influence on each task and the improvement it could provide will be assessed.

Objective 7: Demonstrate the pertinence of the presented solutions in a concrete example
To demonstrate the validity of the solutions described in this thesis and to quantify their influence on the stamping tool development process, a scenario will be built on

the basis of a concrete example taken from a typical application in the automotive industry. Here, all the stages of the process starting from car-body component design up to the definition of the stamping tool will be accomplished for a given component.

1.2.2 Expected Benefits

The expected benefits of the research carried out for this thesis are directly deduced from the challenges in the automotive industry mentioned in section 1.1.1.*The two main focuses are time and quality*. For this thesis, cutting development time means not only reducing time in the car-body and stamping tool development process itself (for the first design as well as for modifications) but also considering the impact of the decisions taken on the following processes.

The expected added value is the *enhancement of development process quality by improving efficiency, flexibility, data consistency, and information exchange*. In the same way as for time reduction, quality enhancement concerns not only the considered process but also any downsteam processes in the chain affected. Thus, as product quality arises first from the product development phase, this thesis should provide solutions to improve final product quality.

The third aspect mentioned in section 1.1.1 was cost. However the topic of cost reduction does not belong to the expected benefits of this survey since costs result from manufacturing considerations, which, in turn, depend on technical solutions defined in the product development stage. The aim of this thesis is to provide methodologies to support the product development process, but not to describe design or engineering concepts for product working and manufacturing, which is a task left to engineers. Therefore this thesis will not address cost reduction directly. Costs could be saved by the implementation of the results of the present work but only indirectly, namely by saving time.

As the aim of this thesis is born from a demand in the automotive industry, *the transferability of its results has a particular importance*. That is why we consider not only the scientific aspects but also practical aspects of the implementation. The expected improvement in this field is the ability of the results to reflect the expectations of the automotive industry and to provide solutions taking industry-specific restrictions into account, for instance, the user bias or system limitations.

1.3 Outline of the Thesis

After this initial chapter 1, the introduction, this thesis is structured in three parts: namely *"State of the art", "Concept", and "Application".* In detail, the first part contains two chapters devoted to the description of the status quo in the scientific area and in the application area. Hence, chapter 2 presents an overview of the topics listed in the following section.

Feature technology is described by the means of feature recognition, design by features, multiple-view features, and features as information core. Meaningful information about the design process chain is given in the form of knowledge-based engineering, expert systems, knowledge acquisition, simulation in the stamping tool development process, integrated CAD/CAD environment, and trends in the design process. The topics of model parameterization and reusability and of master models are addressed and, finally, the topic of concurrent engineering closes this chapter with information about the purpose of concurrent engineering, concurrent simultaneous engineering, organization, tools for concurrent engineering, and the design structure matrix methodology.

The third chapter about the current situation in the stamping tool development process in the automotive industry commences with a general view of the process and then describes the car-body component design process with the related modeling and includes an evaluation of the formability. In the next chapter, the process of stamping tool development is detailed using the design of the die face, the design of the stamping tool, and modeling of the stamping tool. Last, the description of stamping tool production completes the representation of the process chain.

In the second part of the thesis, a concept based on the notion of functional elements is depicted. This part regroups two chapters. Chapter 5 introduces the functional element principle starting with the specification of general characteristics and external influences and closing with how functional elements work. Then the formalization of knowledge and its implementation in functional elements are addressed.

Chapter 6 provides a description of three applications of functional elements: functional features are described in the modeling stage and in the process chain view, the engineering template definition is set out, and integrated elements are presented. Moreover the description of a skeleton to support functional element-based modeling and a technical solution for its utilization are developed. After a synthesis about the common use of functional elements and the skeleton, the contribution of functional elements and PDM systems to collaborative engineering is addressed.

The intention of the last part is to make use of the theoretical principles of the previous part in an example taken from the daily work of designers. In particular, chapter 6 introduces the use of the engineering template development method for the car-body component and its corresponding die face: information acquisition, information analysis, and template definition. Then this method is applied to a car's hood model. A solution to reap the benefit from the simulation and integrate its result in templates is depicted in three steps: simulation information for the component template then for the die face template and coupling of CAD and simulation models.

Chapter 7 presents the engineering template-based stamping tool design and expresses the functional requirements and the problem of user bias. The principle of modular design is described together with the template's input data, its parameterization, and its result. Then template's process information is listed in the form of standards, multiple views, machining information, and documentation. The design detailing completes the use of templates by means of integrated components and functional features. Finally, the result of the detailed model is presented.

Chapter 8 focuses on the improvement of the stamping tool development process. For this aim, the principle of concurrent engineering is used and the design structure matrix of the process is built and optimized. To describe the role of PDM systems in the current topic, the following aspects are mentioned: engineering templates, features, integrated components and PDM systems, and release management. The issue of software customization and system integration ends this chapter.

To conclude the work, chapter 9 looks back at the outgoing situation, provides the synopsis of the thesis, and cites the added value resulting from this thesis. Finally the section entitled "Outlook" offers an overview of the possible topics that could be of interest for further work in the direction of knowledge-based engineering.

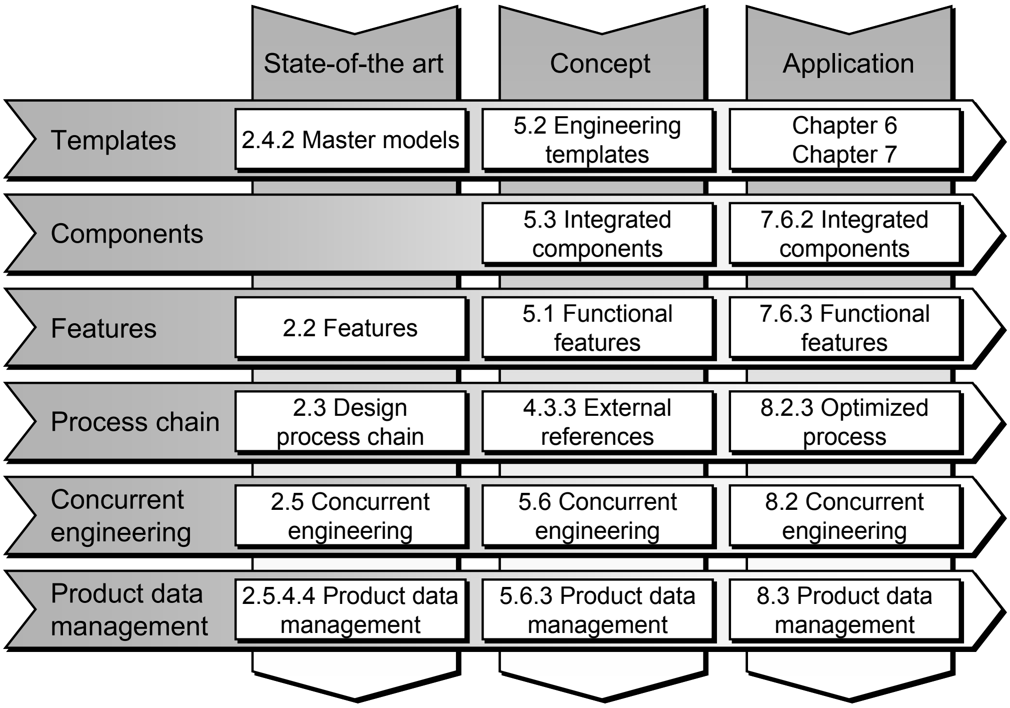


Figure 1-3: Thesis structure.

Most of the key topics of the thesis are mentioned in different places of this document so that there are two ways to read it as shown in Figure 1-3. This diagram gives the references to the sections where each matter is treated. It shows that it is possible to read this document linearly from the state of the art up to the application over the concept or to read it crosswise from subject to subject.

PART I
State-of-the-Art

Chapter 2
Current Research Results for the Product Development Process

2.1 Introduction

This first chapter of the part "State of the Art" gives an overview of the methods and tools supporting the development process as they are represented in the research area. Since the aim of this thesis is first to improve the stamping tool development process, contributions about the following topics have been targeted: technologies and methods supporting product design, solutions to provide a continuous information transfer along processes chains, and efficient organization and working methodologies.

Feature technology appeared to be one of the key contributions in product development. It is described in this chapter under its various forms such as feature recognition, design by features, multiple-view features, and features as information core. Then the topic of design process chain is developed. In this domain, the use of knowledge and expert systems as well as simulation systems play an important role. The evolution of CAD systems involves the development of new modeling methods that are described in this chapter by means of model parameterization and master models. The last section is devoted to the topic concurrent engineering; it presents information about its working and about the methodologies supporting it.

2.2 Features

2.2.1 Introduction

Since the introduction of feature in the 80's, many definitions of features have been given [KrWe-99], from the description of machining operations up to the definition of active elements with designers' know-how. The development of features based methodologies, and tools to support them, began in the research institutes, but today many CAx systems propose functionalities for feature engineering [Erdr-01]. The aim of this section is to give an overview about the various feature definitions.

2.2.2 Feature Recognition and Machining Features

Feature recognition
Following Shah [Shah-91] the first researches on the feature technology had for aim to define geometrical elements being easily extractable and interpretable for the generation of NC programs. Thus this methodology has been called feature recognition. The aim of the feature recognition was to match areas of part model to be machined with predefined features. The working sequence is as follows:

- Searching in the feature database the corresponding generic feature
- Extracting the recognized feature from the database
- Setting the feature parameter values
- Completing the feature geometric model
- Combining simple features

Machining features
The first objective of the machining features was to define volumes to be removed by machining operations but in further studies, features might be an area of interest on the surface of a solid or a set of faces of it.

The modeling of a part was thought of as a succession of removing operations. The research focuses on three machining features: holes, slots and pockets [JHan-96]. The following table resumes the characteristics of these features by giving there name their shape type, cutter type, machining operation, parameters and a picture from the feature with the cutter course.

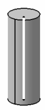
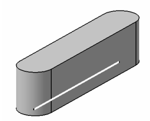
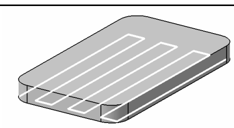
Feature name / **Feature Characteristic**	**Hole**	**Slot**	**Pocket**
Shape	Cylinder	Elongated parallelepiped with rounded ends	Prism
Cutter type	Drilling cutter	Milling cutter	Milling cutter
Operation description	Vertical sweep	Single horizontal sweep	Multi horizontal sweeps
Parameters	Radius, height	Length, width, height	Parameters of the basis profile, height
Representation			

Table 2-1: Machining features (based on [JHan-96]).

The pocket shape can vary, depending on the planar curve that describes its basis. The cutter course hence varies as well. For this there are several strategies, for example, several concentric ways, helicoidally way, back and forth ways. The machining feature recognition is often insufficient to define the process planning. More information is necessary, for example, the volume to be removed. This is calculated from the raw block geometry.

Figure 2-1 shows an example of part with its decomposition in machining features: a slot, a blind hole, a through hole, a pocket, a face, and fillets. As machining features are so-called negative features, some elements that may have been seen as positive, for example, both pins on the top face, are nevertheless obtained by subtraction.

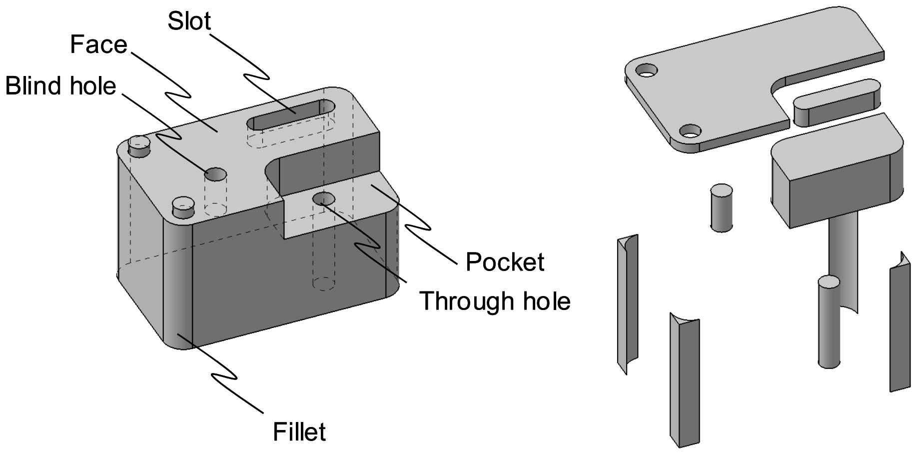


Figure 2-1: Example of a component and its decomposition in machining features.

This method forces designers to think like production engineers. Thus machining features are interesting as a communication form among the designers, but the methodology employed to perform the design work has to fit the engineer's intents. Indeed, design features' shapes are different from machining features' shapes. In many cases, design features (refer to section 2.2.3) have to be decomposed into several machining features. One solution is to generate process-specific features depending on the application they belong to and map them as depicted, for instance, in Figure 2-2, where a design feature "through hole" is matched as three machining features: drilling, chamfering, and reaming [LaHH-02].

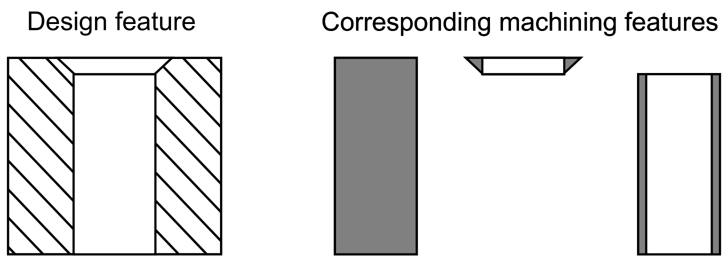


Figure 2-2: Design feature and machining features (based on [Laye-03]).

In [DFGH-94] and [DeFG-96], it is shown that feature-based design and feature recognition are not in opposition but can be joined in an integrated approach enabling designers to design intuitively with features and geometric primitives.

2.2.3 Design by Features

2.2.3.1 Main Principle

Computer-aided process planning (CAPP) involves two types of feature methodologies: feature recognition and design by features. Design by feature does not completely eliminate the use of feature recognition, but reduces it significantly [DiYu-02]. Design by feature differs from feature recognition by the fact that the model is created from the beginning of the basis of features. This approach was developed in the middle of the 1980s by J.J. Cunningham and J.R. Dixon, who proposed an architecture for the design by feature [CuDi-88].

Whereas design by machining feature is based on the subtractions of depression features from a raw block, the creation of a model by design features is based on the adding of protrusion features and subtracting depression features.

For performing design by features, features are classified in a feature library and contain a set of constraints, some of which are set out below:

- Geometric constraints (shape, position, and orientation)
- Topological constraints (constrains keeping the validity of the feature topology, that is, the depth of a blind hole)
- Machining constraints (dimensions and tolerances)
- Interacting constraints (interacting properties between parent and child features).

The main problem in the use of design features is linked to the feature interactions and the model validity. The interactions between features could in this case be managed by a high-level data structure where the relationships are defined (connection, non connection, parent/child relationship). The interactions between the features pairs and the constraint violations are detected and checked during three possible operations: adding a feature, editing a feature, and deleting a feature. An algorithm has been implemented by Ding et al. [DiYA-00] to apply the described working of the feature interactions in a real example. However the definition of feature interactions by designers is still not supported in current CAD systems.

2.2.3.2 Procedural and Declarative Modeling

Two approaches of feature-based design modeling have been identified: the procedural approach and the declarative approach [SBRU-94]. In the former, features are defined by a set of predefined rules and procedures describing the features' working during operations such as insertion, modification, or deletion. In addition to their solid representation, procedural features are defined by two types of parameters linked by rules: parameters that can be manipulated by designers and parameters using rules.

The first feature of a design is positioned in the default coordinate system. Then children features are inserted by using coincidence constraints with the face of the parent feature. The orientation is insured using a local coordinate system. The consequence of such a modeling methodology is the hierarchical structure that is created step by step during the design. There is a chain of relations and a rule order that allows modifications to be propagated only in one direction and insures that it is not possible to create mutual constraints between two elements on the same level.

Declarative feature-based modeling consists of the definition of generic features whose geometry is characterized by spatial relations. Most of the time, these relations or constraints connect low-level entities (points, lines, face, planes) or primitive solids. They define relations such as perpendicularity, parallelism, or coincidence. They are used to describe the inter-relations between elements within and between features. These elements are either visible elements of the features (features edge or face) or virtual element such as planes or symmetry axes. In contrast to the procedural approach, declarative features based modeling may require a constraint solver to deal with conflict problems.

2.2.3.3 Feature Interactions and Model Consistency

In the definition of form feature-based models no difficulty occurs as long as features are inserted disjoint from each other. But most of the time, models are complex so that interactions and overlapping between features may result in feature validity alteration. *There are two types of interactions: volume interactions and boundary interactions, which are, in both cases, the overlapping region of the feature with its closed volumes*. Among the different interaction types, the authors of [BiTe-93] have identified various situations as set out in the following sections.

There are topological interactions where the parameters and the semantic of each features is respected. In this case, the relationship between features does not alter the nature of each feature. This type of interaction is called a topological interaction (see Figure 2-3).

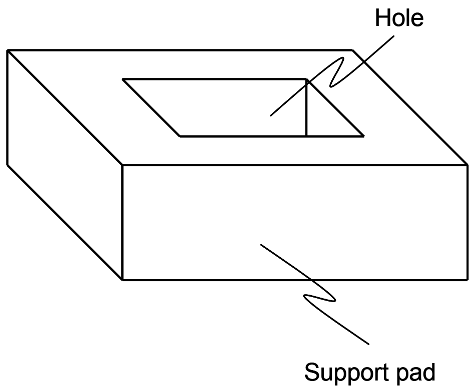


Figure 2-3: Topological interaction (based on [BiTe-93]).

Another situation appears when a feature looses one of its characteristics because of the interaction with another one. It is a transmutation interaction. For instance if a through hole is reaching the face of its support pad it becomes a slot, which was not its primary function (see Figure 2-4).

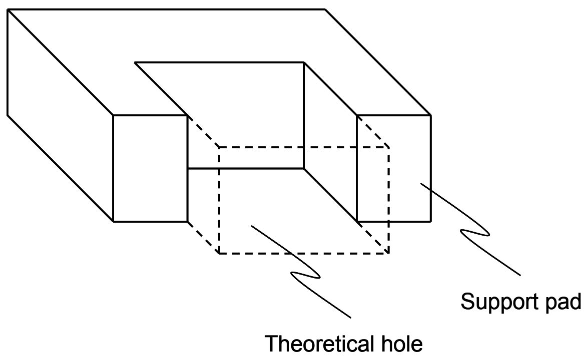


Figure 2-4: Transmutation interaction (based on [BiTe-93]).

The third situation concerns geometrical interactions, where a parameter loses its correspondence to the feature geometry. This is shown in Figure 2-5, where the length and the width of the pocket have no connection with a visible geometrical element.

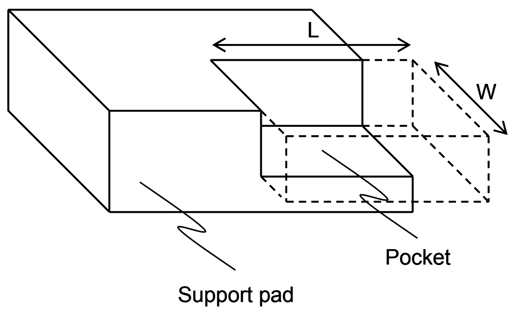


Figure 2-5: Geometrical interaction (based on [BiTe-93]).

If a feature is absorbed by its support, we speak of an absorption interaction. An example is given in Figure 2-6: the model is not consistent as the pocket is fully absorbed in the pad. Another example would be if the pocket were completely included in a bigger one so that it is useless.

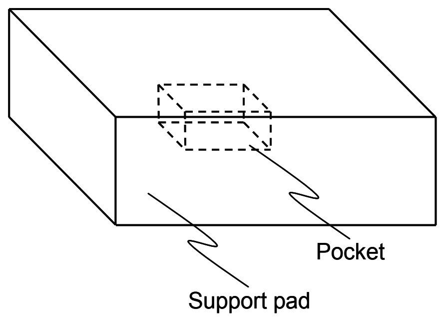


Figure 2-6: Absorption interaction (based on [BiTe-93]).

Feature interactions are analyzed using the description of the feature faces. In fact, features are defined by their faces, which are positive or negative depending upon their nature (open or closed faces). Thus a modification of one of the feature characteristics is detected when a face's attribute changes.

The research done by Bidarra et al. defines various cases of feature interaction and the way they are identified, but what is missing is the way feature interaction should be employed to keep the model consistent. This problem is solved by the implementation of self-validation features [MCOK-97] or architecture for consistency management [OvJa-94]. Rules are used as validation method to check features interactions. They check the topology and the geometry of features by imposing constraints on them. The feature semantic has to be maintained not only during the insertion phase but also during the rest of the modeling operation, and this, independently of the feature creation order [BiBr-00].

2.2.3.4 User-defined Features

In the middle of the 90s, research was carried out to support the definition of features. In [SSJH-94], the authors point out that CAD systems offered few or no possibilities for the users to create their own features. *To avoid the complex task of feature programming, they have developed an application making it possible to develop features interactively*. The problem of CAD systems was that they offer the possibility to parameterize and modify the geometry of features but not to insert non-geometry-related information. In the application, called FROOM (Feature and Relation based Object Oriented Modeling), not only design form features but also manufacturing form features and abstract features can be represented.

2.2.3.5 Features for Styling

In the styling surface modeling area, using CAD systems, there is a need for high-level entities representing curves and surfaces [FoGM-99]. On one hand there is an increasing importance paid to the product's styling and on the other hand the evolution of material and production techniques augments the possibilities so that designers need methodologies supporting the creation of complex surfaces and to generate quickly product variants.

The problem of today CAD systems is that they constrain designers to work with low-level geometric elements that do not represent their intent. Thus form features have been introduced as a group of geometric elements with a functional meaning. Styling features are grouped into two categories: structural features and detail features. The former are used in the preliminary design to describe the important aesthetic surfaces whereas the latter are used in the detail design stage. The objective in this case is merely to generate the geometry quickly. Thus there is no information about the process chain or functional behavior.

2.2.4 Multiple-View Features

2.2.4.1 Features and Concurrent Engineering

In [BNBH-01], the authors present an approach to combine the advantage of form feature modeling and concurrent engineering. It works with a concept of multiple-view features representing all the stages of the product lifecycle. *The first enhancement proposed in this approach is that multiple-view features can support applications of the product development when the geometry has not been fully specified*. The following phases are concerned with multiple-view features: conceptual phase, detail design phase, and manufacturing planning phase.

For the conceptual design three solutions are possible. The first one consists of choosing one previous design and adapting it to the current design requirements. The second one is based on sketches made by designers and converted into 3D model. The third solution is called configuration design and is based on the definition of the product's components and their connections. This one has been chosen for the multiple-view feature modeling.

In the conceptual design view only concept shapes are represented with information such as function, weight, material, or volume. In this model, references from other parts and constraints are used to describe the interfaces between the parts. The product model is hence an assembly model containing a frame and features that specify only the connections between the parts. No other form feature is necessary in this stage.

The detail design phase follows, consisting of specifying the product geometry with form features containing class specific design information. Then the manufacturing planning is done to define the way the parts have to be manufactured and which manufacturing equipment is to be used. More details about form features and manufacturing features are set out in section 2.2.2.

To maintain the consistence between two parts views, the consistence between related elements in both views has to be insured. This is done by using automatic feature recognition and feature linking: links consist of geometrical constraints between faces of different views that are coplanar. The consistence is insured between the concept model and the detail model and between the detail model and the manufacturing model. Each view has its own product model composed of features from specific classes. Thus only information concerned by the considered development phase is present in the model.

Independently of features, the trend in the product development process is going in the direction of multiple-view models. The aim of a multiple-view model is to allow each actor of the process chain to structure the product representation in the way he needs it. Thus the product obtains several views, for example, functional, structural, manufacturing, assembly, or recycling [BrTi-00b].

2.2.4.2 Cost Estimation

Thanks to features, information about the following processes is available in the design stage so that designers can evaluate the consequences of their action on the next steps of the process chain: cost estimation, manufacturing plan, resource planning or measurement [HFRW-01].

In addition to machining planning, the cost calculation is another meaningful use of features as semantic objects along the process chain. The aim is again to give designers the ability to evaluate earlier the consequences of their decision for the following process [FFHL-01a], which represents an important issue as product development is largely responsible for the product costs [LaHH-01]. Cost calculation applications determine the machining time on the basis of the feature's information and other parameters such as machining sequence or material as well as calculation rules. In this way, it creates a feature-based cost model that is directly dependent on the machining time. The advantage of the feature technology in this case is that it guarantees that the cost calculation result is up to date and that it can be updated after each design modification [FFHL-01b].

2.2.4.3 Feature or Object Linking

Technical solutions are needed in order to use features as information core along the process chain. PDM systems may be one solution but this necessitates that features can be represented as individual objects [SAFS-01]. The need for feature mapping is motivated by the fact that there are various kinds of features describing the same product but in different phases. In particular, a feature-based design model and a feature-based manufacturing model are quite different [ShMä-95].

Another approach is feature linking, whose aim it is to accomplish a bi-directional communication between CAD programs and subsequent applications such as process planning. *In other words, information has to be exchanged between various application-specific feature-based product models* [ZiHH-02a].

Features of each application have to be related to others semantically by defining classes in a universal model. Hence, the data meaning of an application can be understood by the others. Then an interconnection has to be insured between the feature instances so that it is simple to find out which features of a model corresponds to those of another. In addition, mapping rules are defined to derive a new set of feature instances from a given one. Such an approach has been developed by J.U. Zimmermann under the name of Universal Linking of Engineering Objects (ULEO) [ZiHH-02b]. It supports the linking not only of features but also of any engineering objects (EO).

2.2.5 Features as Information Core

Some authors describe features as a way to store all the product information in a 3D model [MbSH-02a]. As such, design features represent not only geometric information but also process information. Under this type of feature, there are general features and specific features. The former are features that may be employed in any design context such as holes. The latter are dedicated to one domain of application. Although they may have the same shape as the general features, they are more intuitive to handle for the users (for example, an injection hole) [GeWe-01].

Form features are shape-oriented elements with engineering meaning, which is insufficient for J.C. Borg and F. Giannini [BoGi-01], who propose applying considerations about the product's lifecycle to features. The authors start from the observation that *decisions taken during the design stage have effect across multiple product life-phases such as manufacturing, use, and disposal*. They show that features are defined relating to the way they are produced: for instance, a rib. If the rib has to be machined, its thickness plays an important role during the machining operation. Thus this information has to be inserted in the feature definition. But if produced by molding, the rib geometry should consider other requirements such as the draft angle to facilitate the ejection. This approach makes sense and show that features have to be dedicated to specific application domains.

In [KrRU-94], F.L. Krause, E. Rieger, and A. Ulbrich demonstrated *the need to incorporate non-geometrical information into features, which became semantic objects*. But the presented features were general elements that were not contingent upon the application domain. The authors have identified three types of semantics:

- Design-oriented semantic
- Manufacturing-oriented semantic
- Quality-oriented semantic

Feature definition at FEMEX [OWVM-97]
FEMEX (Feature Modeling Experts) is a working group composed of scientist from universities, research institutes and industrial companies, whose objective was to provide a general definition of the features. For the group "Feature modeling methods and application areas" features have following characteristics:

- Features are the carrier tool of meanings, which should facilitate the designer's task by containing information about their creation, their reusability, and their relationships to other elements.
- Features are structure elements for the overall product development process.

The contribution of the FEMEX in comparison to previous surveys about features lies in the following points:

- Relating to the user's qualifications, features should not be harder to use than common elements of today's CAD systems. Their utilization should be intuitive and accompanied by information for problem solving.
- A feature should check the validity of its environment during its insertion.
- Feature information can be extracted and represented in a desired format.

2.3 Design Process Chain

2.3.1 Expert Systems

Expert systems are tools containing know-how about a given engineering domain and supporting designers in taking decisions. As much work has been done about expert systems for the design of tolling for sheet metal forming, it should be evaluated if they could be introduced in the domain of the car-body stamping tool design.

In [UzRM-91], an expert system is presented for the process planning of sheet-metal parts. The process plan contains the description of the information necessary to manufacture the part, the sequence of operations, and the necessary machines and tooling. The objective in the definition of the process plan is to reduce the production time by minimizing the number of operations and to reduce the complexity of the production dies. As the process is strongly influenced by the geometry of the part to be produced, some rules must be respected during the design stage. In particular, the efficiency of the production depends on the characteristics of basic elements of the part and their interrelations. The experimental system described in this thesis is interesting to save time in the definition of the process planning, but we could imagine that the rules of the expert system would be considered as early as the product design phase. For instance, the relations between the features of a product could be checked during the modeling: angle of the bend, spacing of the holes, distance hole-edge, distance hole-flange, etc.

Note 2-1
The use of an expert system for the definition of the process planning in the production of car-body components seems to be difficult. All the expert systems mentioned in the literature refer to the production of simple geometry parts such as axis-symmetrical parts [SiKA-91]. The rules relating to this kind of parts are simpler than those relating to car-body parts as they are for the most part describable using prismatic geometry [AnKa-94] [CKCK-00] [LiCh-96], being produced in a multi-stage forming system [Tisz-95]. Further, an expert system makes sense in such a case because of the large range of different geometries possible and thus of the various possible production sequences.

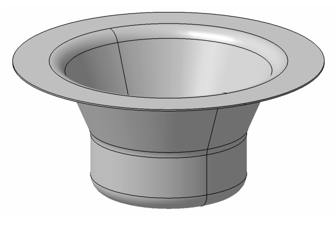


Figure 2-7: Axis-symmetrical part.

Figure 2-7 shows an example where the use of an expert system is significant. In such a case, the rules refer to the ratio diameter/depth or to the angle of the surface relating to the punch direction or to the value of the radii. It is possible to deduce the intermediary drawing passes from the final geometry parameters. In the production of car-body components, the geometry is more complex, making the rules harder to define, and the sequence of the production operations is practically established. Moreover simple rules are not enough to predict and guarantee the behavior of the sheet metal during the stamping operation. The use of simulation systems is more and more diffused in the industry so that evolution will surely continue in this direction (refer to section 2.3.4)

For instance, the simulation of the metal flow over the drawbead is done with a finite element simulation as described in [XuWe-96]. The description of the drawbead involves 14 parameters that cannot be set using only the designer's experience. It is clear that the simulation offers much more precision. However the papers about the subject of expert systems show the necessity to collect the engineering knowledge and to translate it into rules that could be automatically checked as explained in the next section.

2.3.2 Knowledge-based Engineering

In the last years there was a demand of additional information to accompany the product geometry in the design stage [RoTi-00]. Especially data corresponding to knowledge of the various actors of the product lifecycle should be inserted into or assigned to CAD models.

The reusability of existing design is strongly linked with the notion of knowledge integration. The term equivalent knowledge was introduced to designate the behavior properties of an object in design medium. It consists of developing procedures for identical transformations of components by avoiding intermediary decision-making [NaZa-01]. The knowledge has to be captured in existing design of components and used as "design cases". Knowledge is stored in the components in the form of parameters and rules, allowing users to generate component combinations.

2.3.3 Knowledge Acquisition

In [SVCA-94], the authors point out the need to develop user-friendly design tools for the knowledge acquisition and modeling methodologies. The decisions in the process planning result in two types of expertise: calculation and experience [TeVG-02]. Experience is more complicated to implement in a CAx system than calculation, which is of an algorithmic nature. The article establishes that it is difficult to collect the knowledge coming from tables, equations, diagrams, textbooks, handbooks and articles and to code it in the form of rules.

The paper [PaWN-94] introduces a knowledge-based simulation approach for the sheet metal forming. In this study, an example of automotive panel is given. However the aim of the presented system is not to define the stamping tool but rather to work in connection with the simulation software. The role of the system is to suggest to the users the possible way after having identified a problem in the simulation phase.

In addition to knowledge relating to manufacturing, knowledge has been thought as a way to make geometry dynamic. This means that it consists of a set of rules such that the geometry adapts itself to the context [Klei-96] [Klei-97]. Initially, knowledge was left to the user, and surveys were then performed to define dependencies between geometric elements (particularly between features). It is now possible to insert them into models using standard functionalities of CAD systems.

The need to reduce development time by automating best practices is motivated by the following factors:

- Engineers spend time redesigning each new product from the ground up, despite similarities with existing designs.
- By routine work, errors involved by the monotony of the work cannot be avoided.
- Engineers do not have all the required know-how in mind, as they may not own the right information or may have deleted it [Brow-04].

2.3.4 Simulation in the Stamping Tool Development Process

2.3.4.1 Formability Simulation

Due to the necessity to reduce the product development time and to minimize the costs, simulation plays a key role in the design process. Its goal is to assure the quality of the product definition as soon as possible and to avoid time losses due to the iteration loops from the product definition to the product test (trial-and-error method).

In the die design process, the formability simulation has been established for several years and other types of simulation are coming, for example, the simulation of the subsequent operations or tooling line simulation. In [Piet00], the benefits and the working of a formability simulation are mentioned. It aims at making component geometry free from errors, fulfilling the quality requirements, limiting the costs, and defining the production operation sequences. The simulation is only one task of process planning (for more details, see section 3.4.1), which is subdivided as follows:

- Die face geometry development
- Process simulation and result evaluation
- Data feedback to the CAD system

In order to simulate the stamping operation, a first draft of the die face has to be created. Both operations are done in the same system. The modeling of the die face consists of developing the flanges, extrapolating the component geometry, and defining the blankholder and the addendum surfaces. In particular, the addendum surface is based on a set of profiles representing the connection between the component geometry and the blankolder. Users can set the parameters of each profile to influence the addendum's shape, which is generated by the system in the form of a meshed surface.

The formability simulation gives the opportunity to evaluate the feasibility of a car-body component and to integrate notions such as springbacks and flattening effects after the stamping operation [Schu01] in the design phase. The simulation result gives a good view of the future sheet-metal behavior but there are some aspects to improve:

- Raising of accuracy
- Integration of new forming methodology and new materials
- Link to other CAx applications

In addition, K. Siegert demonstrates in [Sieg-99] that simulation, despite the progress it represents, cannot replace the production of a stamping tool prototype. The simulation enables several tool constructions to be avoided, but some modifications are still necessary after the part test phase. These modifications are not necessarily due to a problem of the stamping tool, but may be justified by the part's behavior during the vehicle try-out.

Note 2-2
Apart from the coupling of simulation and CAD software, there is also the problem of the validity of the simulation results. Although simulation software gives significant previsions about the component quality, they are today not able to hold with precision all the physical problems such as the springback effects [Besd-99]. That is why the production of prototypes for testing of real parts cannot be disposed with, even if their number has been reduced over the last years. As a consequence, if the part has to be modified, the tool has to be modified as well, so that the goal of an efficient design process chain is not only to reduce the number of modifications but also to make it possible to deas with the modifications as quickly as possible.

Nowadays there still exists a separation between the stamping simulation (or any type of other simulation) and the CAD modeling stage. Indeed, as explained above, the FE geometry created for the simulated part must be remodeled in the CAD system. That is one of the reasons why designers try to model the die face as late as possible: the design of the die begins only when many iteration loops have been run in the simulation system and when the die face geometry is fully defined [RATH-99].

2.3.4.2 Simulation for Next Steps

The formability simulation is traditionally used to validate the part design: by performing the simulation, designers can identify the area were the sheet metal is too thin and modify either the part geometry or the die face geometry to deal with this problem. The weak point of today's methodology is that, as soon as a critical area no longer exists, the part is considered perfect. That means that the nominal part geometry will be used in the further steps. For example, the part models are employed in the crash or acoustic simulations [RoHe-99].

However, although the result of the formability simulation validated the part geometry, the real part will not exactly match the virtual model. After the forming operation, there are residual stresses in the part due to plastic deformations. There are also slight deviations from the ideal geometry due to springback effects and thickness differences resulting from the sheet-metal deformations. All these factors have an influence of the part's behavior in the future car assembly, which is why it is important to use the formability simulation result as input for the other simulation disciplines instead of using a "perfect" CAD geometry.

Simulation software applications work with finite element models (FE models). What is missing is the interface between them, so that the result of the formability would be directly used in the next simulation steps. In particular, not only the FE model would be necessary for the structure simulation, but also information about the strain distribution and thickness deviation.

For the issue of data conversion and software interfaces, the expected solution is not the use of a single application that could perform all the design tasks but rather to optimize the data exchange between the applications, particularly between CAD- and FE-based simulation systems [KrWö-01]. In fact, it is a common requirement for all the CAx systems of the process chain to exchange data with more information than the purely geometrical [StTh-03]. *Above all, parametric model exchange data are needed so as to enable modification in each step of the process chain*.

2.3.5 Integrated CAD/CAM Environment

2.3.5.1 Continuous Process Chain

In most of the design environments, the question is not the one of the implementation of a CAD/CAM/NC process chain anymore, but rather how the data exchange works. Indeed the companies have already integrated the idea of a continuous process chain, but the problem lies in the data interfaces [Kloc-02]. For instance, the data transfer between CAD and NC systems is an important point both in view of data quality and also in view of model conversion. In [Dieh-98], *L. Diehl shows the importance of integration of machining considerations in the design phase*. Only small modifications such as round corner or draft angle modifications may result in a great difference in the machining costs.

2.3.5.2 Machining Process Sequencing

As mentioned, machining features contain information for the definition of the machining operations, but they can also be a support for the definition of the process planning. The operation sequencing is the task of defining order of operations to product a part with the objective to minimize the sum of machine, setup, and tool change costs [WCFS-03].

In [Laus-03], the authors demonstrate that feature technology makes it possible to start the preparation of machining planning before accomplishing the part design. *The machining should then begin as soon as the design is completed. This is realized by possibilities offered by the model associativity used in a concurrent engineering schema.* The NC program can be automatically generated from the CAD model, using feature attributes. The optimal manufacturing process as well as the required milling machine are then defined with help of predefined rules and conditions. As a result, the machining time is reduced and errors are avoided.

Machining features are today used in CAD/CAM systems offering a continuous and integrated solution for the product modeling and the definition of the process planning. Among their advantages, we can underline the following:

- Automatic recognition of machining elements
- Automatic sequencing of machining operations according to given rules and criteria such as the following:
    - Grouping of machining operations done by the same tool
    - Grouping of machining operations on the same side of the component
    - Reducing of machine table rotations
    - Sequencing of the operation types (pocket before holes, etc.) [Fetz-01]
- Graphical simulation of the NC program

However, the overall geometry cannot be considered in the feature recognition [Fetz-00]. A stamping tool, for example, contains free form surfaces that do not belong to the frame of prismatic geometry.

In [KaSf-01], the importance of flexibility in process planning is underlined, not only because of the inevitable design modifications but also because of requirement modifications for the manufacturing stage itself. These causes of possible modifications are listed below:

- Manufacturing plan changes due to new subcontractors, replacement of production equipment, or transfer to a more attractive manufacturing plant location
- Technological innovations
- Upgraded specifications (production volume, new performance, new standards, etc.)
- Additional marketing requirements because of customer feedback, for example
- Environmental restrictions
- Need for improvement of the product cost and quality

These conditions cause alternative manufacturing methods to be quickly generated and compared. The method used by G.J. Kaisarli and M.M. Sfantsikopoulos to define the manufacturing process on the basis of the part model is called feature-based process conversion. In this approach, form features are converted into "process compatible features", which is particularly interesting, since it shows that form or design features do not have to be opposed to machining features. In addition, it indicates that form design features can be converted into different process feature types, depending on the way the part is to be manufactured (machining or casting and machining, etc.). Thus designers do not have to necessarily to know the intended production process during the product design phase. After the model conversion, the applicable process and additional design guidelines of a knowledge-base are considered.

2.3.6 Trends in the Design Process

In [RoHS-97], the authors identify the weak points of the process chains in the industry. Although their work dates back to 1997, many of the issues cited then still remain relevant today:

- Unidirectional information flow
- Information losses because of data interfaces
- Lack of knowledge exchange along the CAx process chain

To enhance the efficiency of the process chain in view of delays and product quality, design processes have to progress to integrated design processes where an actor's decision would be made according to constraints given by the other actors [TiTo-93] [BrTi-00a].

Contrary to what is usually done in the development process, the information flow has to become active. As W. Dankwort explains in his paper about long-term CAD/CAM strategies [Dank-00], *information exchanges, made of file, paper or physical model transfers should evolve in the direction of CA-conferencing and multimedia applications*. The future strategies should be based on open architectures that are hardware independent. Thus to integrate the subcontractors using various systems into projects supported by the car manufacturer, exchange formats such as STEP (Standard for Transfer and Exchange of Production data) may be preferred to native data formats [DKLM-97].

In [Ovtc-01], J. Ovtcharova gives an overview of the current situation in product development based on CAx systems. There are enormous data redundancies as well as poor data synchronization, user bias issues, and high costs for non-value-added work. *Thanks to the introduction of digital mock-up (DMU), various views should be integrated in a 3D model to enable a central representation of a product with all the necessary information.*

In [WaMi-00], F. Wang and J.J. Mills depict the need in the product development process and explain the necessity to manage product data models. But, for them, the use of exchange formats is not sufficient because of the need for further information such as parameters and behavior. Further, the geometry of models described in exchange formats is fixed so that they have few reusability abilities. *What designers need are parametric and constraints-based models that can be customizable to generate product families.*

In particular, for stamping tool development, efforts have to be concentrated on the CAE techniques to assure the desired product quality. Knowledge-based systems, databases, intelligent design systems, simulation, and visualization tools are needed. *The aim is to make a virtual product representation where all the following process could be represented in real time and effects on the parameters could be optimized dynamically [WaSc-97].*

Apart from technical considerations, the evolution of the development processes and their complexities dictates that care be taken in view of other adjacent problems:

- User bias: Product development is not only a technical process but also a social process with relations between persons and organizations [Minn-91].
- Designers' qualification: The qualification of designers is defined as the knowledge, the ability to solve problems and to learn; but other points are also to be considered such as personality, initiative capacity, and social capacity. The designers' training is very important if new methodologies are to be implemented. It begins in the universities, where the students get education and training about the modeling process chain [DiTT-01].
- Intellectual property: The integration of suppliers in the process chain makes it necessary not only to exchange data but also to manage the intellectual property. This is motivated by the aim to give the supplier the same design and modeling methods as the client while simultaneously protecting its know-how [Gill-01].

2.4 Model Parameterization and Reusability

2.4.1 Model Parameterization

2.4.1.1 Skeleton Model

The idea of a skeleton model is based on the observation that a new product is often comparable to a previous one but with different dimensions and element position [BäHa-01]. In fact a product geometry can be defined on the basis of reference elements such as points, axes, planes, and surfaces. A set of reference elements may be used on the component or assembly level and can be reused for future designs. It is called a "skeleton model" and represents the foundation of the geometry positioning.

In particular, skeleton models can be utilized as the master element for features. The adjustment of the feature's position is made by modification of the skeleton model whereas the shapes of the features are driven by the feature parameters. For a product composed of a part assembly, there is the possibility to define several skeleton models with a hierarchy: a skeleton for the product assembly, skeletons for sub-assemblies,, and a skeleton for each part. Each skeleton refers to elements of the skeleton placed above it (top-down approach).

2.4.1.2 Function Model

In the literature it appears that the idea of skeleton model and of function model are often linked. This is due to the fact that the modeling with skeleton leaves the concept of pure geometrical modeling to suggest a method based on the definition of framework for a technical solution.

In [ToBT94], the authors describe an approach combining skin features and skeleton features. Skin features are the ones giving a technical solution to the required function and skeleton features connect skin features by adding material between them. The aim of the method based on skeleton features and skin features is to make one or more preliminary designs, allowing designers to take a decision for the final product design. This should work in a simultaneous design environment where the various actors may intervene earlier in the product definition. The advantage is that this supports designers in the domains where they do not have the necessary know-how.

Note 2-3

It is not easy to make people work together because of static processes, thus the question is whether there would be the possibility to involve all the actors not in the design stage, but rather to give designers the information they need. For example, features could contain technological know-how about the following processes supporting the designers by taking their decisions.

In fact, skeleton features as they are described here are a different notion than the skeleton models described in other sources. Skeleton features are not a support for the skin features but a connection so that they should not be created first. Both skin features and skeleton features define the geometry of a part but have different functions.

2.4.1.3 Constraints and Relations

A considerable advantage can be taken from the CAD models when they become active. It means when there are relations and constraints between the elements of a model, it is rationally updated in the case of modifications. Today's CAD systems integrate the notion of constraints facilities but their possibilities are mostly not as efficient as expected, so that research is being done to develop constraints solvers.

The shaping of a product model can be defined by a parameter's values. Relationships between parameters are realized by constraints. In this way, the geometry is arranged by solving a set of constraints. The solution of a constraint set may differ, depending on the nature of the modeling system: we differentiate history-based design systems or variation-based design systems [Rude-98].

In a history-based model, the constraints are evaluated in the order they were created. For this reason, some parameter values are locked while others are free. Free parameters can be changed and the values of other parameters calculated based on them. A variation-based model allows the user to change any parameter value desired. Once a parameter has been manipulated, all the constraints are evaluated and solved simultaneously. Work such as presented in [LaLM94] depicts systems representing the relationship between attributes of an object and maintaining the consistency of the model.

Note 2-4
The weakness of such systems is that they make it mandatory to define the relations and to describe the complete structure of the part. Thus it is difficult to foresee such a method for a designer's daily work as the complexity of the relationships in a product is too important to expect such effort from designers.

2.4.2 Master Models

Object-based modeling is a methodology that has gained more and more importance in the last few years so that CAD systems are now offering possibilities to adopt this way of working. Object-based modeling means describing a product with structured objects having interrelations.

Objects have internal properties whose access is restricted by the defined interface. Thus only specific parameters can be changed, with specific methods. The second characteristic of such objects is the ability to transmit their properties to other objects that can perform an action on it [Hoch-03]. The identified advantages of object-oriented modeling are given as follows:

- Adaptability and reusability of accomplished models
- Verification of conformity to standards
- Simple check of complex structures
- Error avoidance
- Time saving

Objects have to be simultaneously flexible and stable. Above all, the relations between the components of an assembly template should be carefully considered. The issue is to define the consequence a modification has on other elements of the assembly [Hoch-04].

H. Waltl describes in [Walt03] the advantages of a "Master model" in the stamping tool development process. *A master model is a complete detailed model of a previous design, which should be reused as basis for a new design*. With this methodology, modeling time is consequently reduced and represents only 50 to 70% of a conventional new construction. If there is no existing model corresponding to the new project, the design is made from base modules that are selected and combined. For this aim, modules and components have to be accumulated and made available in a library where the designers can choose the required items for a design. The estimated time for such a work is almost 60 to 90% of a conventional design.

The reusability of components or parts of the components calls for a greater degree of standardization in the design and production processes. First of all, each new component production is very expensive, even if the design takes only ten minutes

[Konst-00]. Second, the reuse of existing designs does not have a negative influence on the quality of the final product. On the contrary, it can contribute to quality improvement as it allows designers to concentrate on the pertinent part of the new design, where there is more potential. The standardization does not mean that all the products look the same, as infinite base component combinations are possible and only the significant product characteristics for the customer should be modified.

C. Weber and H. Werner describe in [WeWe-01] an object-oriented modeling method that has been implemented in a system called Ligo. This approach proposes two enhancements for product design: the integration of design know-how and the integration of functionalities of various CAx systems. The principle of modeling with Ligo is to reuse objects (or modules) that are frequently needed. The modeling of generic object classes is a task that is time consuming but justified by the later reduction in design time.

2.5 Concurrent Engineering

2.5.1 Purpose of Concurrent Engineering

2.5.1.1 Introduction

The idea of concurrent engineering (CE) has been developed in order to give designers the possibility to evaluate their designs as soon as possible. In a traditional engineering methodology, the development of product variants and their evaluation is a repetitive job that represents a great waste of time. Thus there is a need for methods supporting the evaluation of product variants and eliminating routine jobs.

Definition 2-1: Concurrent engineering (adapted from [Guna-98])
Concurrent engineering is a systematic approach for the integrated, simultaneous design of both products and their related processes, including production.

J.R. Hartley introduces a general approach toward the concurrent engineering problematic in his book "Concurrent engineering, shortening lead times, raising quality, and lowering costs" [Hart-92]. He shows the necessity to create products that better meet the customer’s requirements. That means that the industry not only has to increase its performance but also needs to shorten the time to market and improve quality.

The Japanese model consists of starting the advanced engineering phase earlier than its American and European competitors. The consequence is that engineers work closely to the concept stage and have to be aware of what happens there. If they did not, they would create an obsolete design that no longer suits the concept. A further consequence is that few design changes are made late so that the quality problems resulting from design changes in the few weeks before start of production are avoided. Leading this way of working, Japanese industries have shortened the time to market to 36 months in the automobile industry. Yet, this time has to be reduced even more.

2.5.1.2 Extra Cost Now Saves More Later

The globalization of manufacturing and the competitive environment have clearly increased in the last past years and have reached a serious and sustained high level of intensity. The author of [Mill-93] points out that the design stage is an early stage of the overall process and does not carry most of the costs. He showed that the design costs represent only 5-8% of the costs, while having a very important place as it can entail massive expenses in the following stages. This problem is the issue of design changes that have a grave impact in terms of cost. Yet the product lifecycle price is fixed during product development. This can be explained by important decisions made in upstream of product development, where the functions of the product and their technological solutions are defined.

Modifications may have very different consequences depending on when in the process they are made. In the traditional design approach, late modifications have expensive consequences as they affect several departments. As presented in Figure 2-8, generally insufficient effort is invested in the early phase of the design so that changes become necessary relatively late. In these cases, after start of production, problems are detected and changes have to be made, generating unreckoned-with costs.

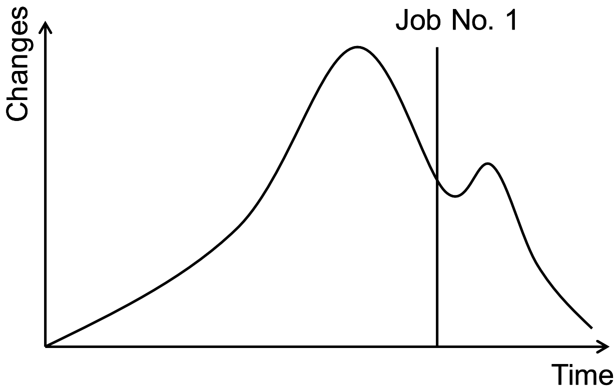


Figure 2-8: Changes in the traditional engineering process.

In the concurrent engineering approach, most of the changes occur earlier in the design process so that the product's characteristics can be validated before start of production (Figure 2-9). The number of early changes is more important than in the traditional approach and generates more costs. However, they are not as expensive in the end as changes happening later since they avoid strong modifications in the production lines.

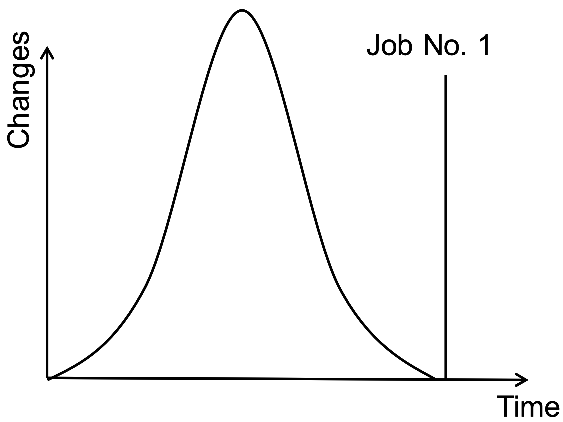


Figure 2-9: Changes in the concurrent engineering process.

As example, a change appearing during the design test costs 10 times more than during the design stage. Analogously, it will cost 100 times more if it takes place during process planning, with this figure soaring to 1000 and 10000 during pilot production and volume production, respectively [Mill-93].

Finally, the problem of design modifications cannot be avoided. That is why it is important to deploy systems facilitating the modifications [FeKr-03]. In addition, methodologies are necessary to make flexible and variable designs.

Top-down approach vs. concurrent engineering
The top-down approach taken in traditional engineering is composed of several stages that, in turn, are composed of others and so on. This way of working implies many interfaces or breaks between the various stages. This top-down approach is also linked with a corresponding technically oriented way of management. That means that managers use the knowledge they have acquired from experience in lower positions. The problem then is that the technologies evolve too quickly and that managers need to follow these evolutions. Additionally, managers have to deal and negotiate with each other to define a suitable compromise for the final product. In a concurrent engineering approach, a process-oriented management has to be introduced where management can define the configuration and variants of the design, having an overview of the process.

2.5.2 Concurrent Simultaneous Engineering

Since the 1990s, concurrent simultaneous engineering (CSE) has been recognized as a strategy for product development. The reactions to organizational modifications often meet with reserve from the people involved since they do not necessarily see the individual opportunities. However, it is essential to adapt the organization to the new situation [BeSt-96].

CSE is a whole strategy that cannot be introduced without taking care to all the parameters of the process. The organization and the structure were the first developed points; another important one is the information technology (IT). The new IT structure should not only be an information exchange means, but rather a platform to support the CSE logic [Kess-96]. CSE could be defined by these three key words: parallelization, standardization, and integration as detailed below.

Parallelization
Parallelization controls the access and distributes the data and information, allowing a multi-project-management. Systems must be flexible and adaptable to the new design requirements.

Standardization
Following points are targeted by the standardization:

- Reusability of the data and standard functionality and interface for different tools
- Providing libraries for project tasks
- Reusing existing components
- Providing functionality to model processes and their data interdependence.

Integration
Various kinds of users and applications have to be supported and integrated in the same project. The automobile industry has understood for a long time that the product has to be designed with a view to manufacturing and assembly. That is why product designers are supported by feasibility engineers: their task is to define how the product could be made, which material is to be employed, and whether the product can easily be manufactured. However, the necessary modifications to be brought to the product definition in sense of feasibility are still difficult to implement as they involve enormous effort when the advancement of the project is significant.

The problem of design to cost is an approach that belongs to concurrent engineering methodology. While calculating the cost of the product after the product has been fully completed, the calculation should be updated at any time and done in parallel to other tasks. This is possible when designers have taken the manufacturing cost and the assembly cost into account during the functional design. The author of [Slee96] proposes a system to help the user to manage these data by structuring them.

Reuse of existing design not only reduces design time but also has a tremendous impact on manufacturing. The problem is rather to capitalize the company creation and to make it possible to reuse a design. In so doing, standardization will increase, making it possible to reuse process, resources, and project and cost data (additionally to geometrical data). CSE hence consists of parallelization and also of avoiding new activities.

2.5.3 Organization

The arrangement of the tasks and the introduction of concurrent engineering can only be successful if accompanied by the definition of the cooperation between the different process activities: this also means *taking the structural organization of the company into consideration* [TiBr-00]. Concurrent engineering calls for modifications in the work organization; yet the whole company does not have to switch to this new organization - it can be done step by step. Second, particular attention has to be paid to the people involved. It is necessary to change the way people think and the corporate culture. And this seems to be the greatest problem. One of the answers is training. Companies must train their people to use the essential features of CE and take a practical approach to its working. Further, the implementation of CE has to be promoted by management responsible for the project and pilot projects have to be led.

The organization of work and personnel planning are aspects that have been underestimated in previous research although they are a key for success in CE in the sense of quality and time potential. The optimization of the work organization begins with the analysis and the definition of the work. The following aspects are examined:

- Environment conditions
- Information flow
- Cooperation
- Process
- Realization
- Tools
- Performer

- Location
- Time [Stah98]

The result of this analysis is that the organization of the work should be supported by a methodology that defines the elements of a work system and the relationships between them. The method yields a better comprehension of the tasks and defines relations that were already known but also defines new relationships that are necessary for the stability and the security of the process.

2.5.4 Tools for the Concurrent Engineering Approach

2.5.4.1 CAx Systems

Concurrent engineering is not only an organization of work but has also to profit from the technologies and communication means. In this context, CAD/CAM systems have a big role to play because they allow the parallelization of the work and the data transfer [Hart90]. Above all they give the possibility to visualize, simulate and evaluate the product's characteristics. The implementation of concurrent engineering is strongly influenced by the software's possibilities.

The CAD system is actually in the center of the product development process so that design for X tools have to work on the CAD model in order to provide designers feedback from manufacturing, cost and quality analyses. A higher level of information degree can only be supported by intelligent CAD systems. CAD systems yield digital information that can be directly used by other systems, for example. machining and simulation systems [AlFa-97]. For example, it is possible to mill a cavity for a mold or a die directly from the CAD model data. Simulation loops can be run once a rough concept is ready. The simulation can hence start to validate a design and help to generate alternatives before the complete design is finished.

2.5.4.2 Knowledge-aided Engineering

Tools propose a common shell for the product development where CAD systems, simulations tools and virtual prototyping technologies should efficiently interact [MRSC-01]. The information needed by the product designers is integrated in a Knowledge Aided Engineering (KAE) system. The generation of a product model in a KAE model works in two times: at first the generative model is created, then experts of various disciplines intervene to insert design intents, relationships, engineering and manufacturing rules and best practices. To generate several product variants or to make a new project, the design specifications have to be changed by the user. The model will be automatically updated by tacking into account all the rules inserted by the experts.

2.5.4.3 Common Database

In the future, concurrent engineering should rely on a database used as a common platform for all the people taking part in the project. In other words, the database will support the work of the CE team, allowing each worker access to the latest data from the concept to final design. The data should be available as the following:

- Design data for product engineering and component suppliers
- Functional design specifications for specialist suppliers
- Manufacturing data for manufacturing engineers

- Full specifications for cost analysis
- Specifications in product terms for marketing

2.5.4.4 Product Data Management

Innovation focuses not only on the technology area but is also a question of process and organization, which can be strengthened by the use of Product Data Management systems (PDM systems) [Uhly04]. The innovation in product design depends more on the improvement of existing products than on wholly new inventions. In other words, innovation is mostly based on the present know-how [Stap-02]. In this context, PDM systems are a part of the innovation process since they support standardization.

Definition 2-2: Product Data Management
Product data management is the discipline of designing and controlling the evolution of a product design [CrPS04].

The functions of PDM systems are the following:

- Data archival
- Documentation management
- Workflow and process management
- Product structure
- Configuration management
- Classification and retrieval
- Communication [ZhCE99]

The association of CAD and PDM systems helps to shorten the product development cycle [Luby-01a]. *PDM systems provide access to engineering data and promote communication and information exchanges* [GPSM-01]. In addition to CAD data, the product definition cycle involves much non-geometrical information: bills of material, notes, process records, simulation data, functional specifications, change orders, etc. Thus rapid product development calls for an efficient information access [Luby-01b]. As many actors and various systems take part in product development (CAD, CAPP, CAM, CAE), a common platform is needed to manage all the product lifecycle views. PDM systems provide a collaborative environment by means of product and process lifecycle management. PDM is not merely a solution to integrate systems of different types in the same platform, it is also a means to support systems of the same nature (for instance, various CAD systems), which is a great advantage for supplier integration into the product development process [ScOt-04].

In the last ten years, the advantages of a continuous process chain from product specification up to maintenance have become clear and there are commercial systems to deal with it. However such a solution has not yet been established in most of the companies due to the complexity of the products and processes. Generally, processes are considered step to step and not as a whole [FlHS-04]. The product development process is largely dependant on product knowledge and manufacturing knowledge. When both are well known, that is, the case of the development of a product that has similarities to an existing one, PDM systems are one of the solutions for the management of the design process workflow and for the storing of parameterized models [NoTi-04].

Information flows are typically oriented from product development up to production, but it is also pertinent to establish an information flow in the opposite direction. The motivation for this is to capitalize on the experience of downstream process steps to make it available for upstream ones [RKFM-04]. In this way, information pertaining to assembly or production planning is recorded and fed back in the design step.

PDM and stamping tool design
In contrast to the car-body design process where one designer is assigned to the development of a component, the stamping tool design process involves several designers. Also, it takes place in the middle of the whole process chain of the car-body development and production, and it is thus dependent on the previous data.

The main problem is that modifications of the car-body component specifications – or the creation of a design variant – have to be reflected in the stamping tool design although this may already have started [ReZi-04]. In consequence, the following points have to be considered to define the role of a PDM system in the stamping tool design field:

- Complexity of the data flow
- Number of designers involved
- Change management
- Variant management
- Data actuality
- Deadlines and data delivery

In the process chain of auto-body panel production, the modeling stage takes on special importance because of the number of designers who are involved [CRMW-02]. Some key points have to be considered:

- Consistency of the data in the semantic and syntactic views
- Information reliability
- Efficiency
- Information security

2.5.5 The Design Structure Matrix Methodology

As concurrent engineering is also based on task parallelization and the cooperation between disciplines, it is important to define the relationships between activities. Thus methodologies have been developed in order to clarify these relationships and to reduce design time and costs. *Most of the methods are based on the segmentation of the design process into elementary activities, the analysis of these ones and the new structuring of the process* [PaMa-01]. One of the methods is the design structure matrix (DSM), which consists of building a matrix representing the interdependencies in a process [Tyso-01] [LSFX-03]. It is a quadratic matrix where the tasks are entered in the rows and columns. A cross in the intersection means that the task of the corresponding column needs information from the task of the corresponding row as illustrated in Table 2-2.

	Activity 1	Activity 2	Activity 3	Activity 4	Activity 5
Activity 1					
Activity 2			X		X
Activity 3				X	
Activity 4	X				
Activity 5			X		

Table 2-2: Design structure matrix (based on [LSFX-03]).

As Table 2-3 shows, three relationships can be identified. First if two activities are performed without connection to each other, they can be made in parallel. In this case there is no cross in the intersection of their row and column. In the second case, the activities are sequential: one of the activities needs information from one other. Thus, the first activity is influenced by the second. In the third case, both activities interact so that the second activity is based on the first one, but the first also needs information from the second to be completed. It is a coupled schema.

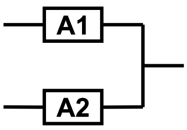
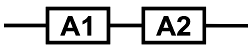
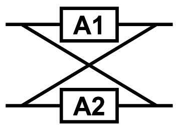
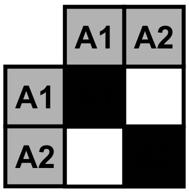
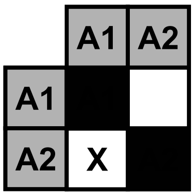
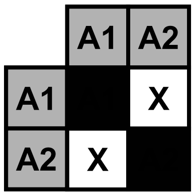
Relationship	Parallel	Sequential	Coupled
Graph representation			
DSM representation			

Table 2-3: Sequence types (based on [BiTe-93]).

The DSM analysis gives a clear view of the interdependencies in the process. It serves then as the foundation for the definition of an optimized concurrent engineering process.

2.6 Conclusion

Originally the topic of this work was feature technology applied to stamping tool design, but the present part of the thesis has clearly shown that *feature technology is already mature and deployed even if there are still some improvement potentials, especially by the implementation of feature technology in operative areas*. *As a consequence, the subject has been widened to the definition of any technology improving the product design and enabling a continuous process chain. That is why* *master models, knowledge-based modeling, and concurrent engineering have been mentioned.*

Concerning features, the improvement possibilities are quite limited, but there is the possibility to develop solutions regrouping the benefit of process-oriented features, while simultaneously supporting the product designers themselves. Indeed these people are the first actors in the process chain and features have to match their way of thinking. Yet, as features remain up to date and will not be replaced by other technologies, we therefore need to define the relationships between features and other new technologies.

It appeared that master models (or design templates) represent the trend in the companies but little work concerning this subject is available. Further, templates only contain geometrical information aimed at the optimization of the modeling process. Global solutions to enhance the efficiency of the whole process chain are missing as well interfaces between models and applications.

The section concerning the concurrent engineering methodology depicts its goal and its theoretical principles. Concurrent engineering is not only an organizational problem but also a technical problem. That is why it is essential to define how concurrent engineering should be supported. Many surveys identify solutions in the form of data bases, data exchanges, and PDM systems, but it is also necessary to define the designer's tasks and their sequencing, with consideration, first, of the concrete engineering data form and, second, of the systems employed.

In conclusion, this chapter has shown the necessity to develop methodologies for an integrated design process. Efficient design solutions, associative methods, or process-oriented methods should not be isolated contributions but should be integrated into a global approach. That is, the issue of integration of manufacturing information in the product design or continuous CAD/CAM process chain has often been treated, yet other activities of the process also have to be considered such as simulation, which will be a key to future process evolutions.

Chapter 3
Current Process

3.1 Introduction to Process Chain Analysis

Definition 3-1: Process
A process is a series of actions that produce a change or development [Coll-86].

Most of the time, process chains are extremely complex and may not look rational for an external observer. Their complexity may be explained by several factors:

- The history of the company: A process chain does not come into being in one step. It is the result of the history of the company and the evolution of the previous processes. Moreover the existing processes are not sufficiently flexible to be adapted quickly so that some situations persist although they seem to be illogical.
- The structure or the organization of the company: A process chain is influenced by the organization of the company, which defines the tasks to be achieved in the various departments. For example, it is possible to distribute the tasks according to personnel skills so that each department has its own core competency, for example, design, simulation, planning. Or it is possible to organize the process by application domains. It this case, a department is concerned by one category of product development and employs various skills to reach its goal.
- The qualifications and knowledge of the engineers: The skills of the employees influence the processes in the sense that the processes are often based on what the people are able to do.
- The system selection: The processes are contingent on the systems chosen for the product development as their functionalities define the tasks to be achieved. For example, the possibility to simulate any process can suppress or reduce the necessity to carry out real tests.
- The strategy: The strategy of the company causes some processes to be promoted in comparison to others so that they are better optimized than the others.
- The suppliers: The fact that suppliers play a role in the process changes the task organization and the information flows.

All these elements yield the result that process chains are sometimes not as efficient as they could be and that they are not easy to modify. However, an analysis of the currently deployed process chain in product design is necessary to understand the sequencing of activities and to deduce the potential of new methodologies and thus of new processes. *The aim of process chain analysis is to evaluate the chain's flexibility and efficiency and to extract the requirements for future improvements of the process.* This may be done by several methods consisting of identifying the tasks, their time, the information flows, and the structural organization of the company. We have chosen a method that focuses on the task and the information flow description: the SADT (Structured Analysis and Design Technique) based on the IDEF0 (Integration Definition for Function Modeling) technique [FISP-93].

The IDEF0 technique bases on boxes representing activities or functions whose names have to be composed of verbs or verb phrases. Arrows represent relationships between the activities. Arrows entering boxes on the left side represent inputs that are processed by the activities and transferred to the right side as outputs (see Figure 3-1). On the top and the on the bottom, arrows represent controls specifying the conditions and mechanisms necessary for the performance of the activities. The method is based on diagrams containing several activities. It decomposes each activity by detailing it in a new diagram at a lower level.

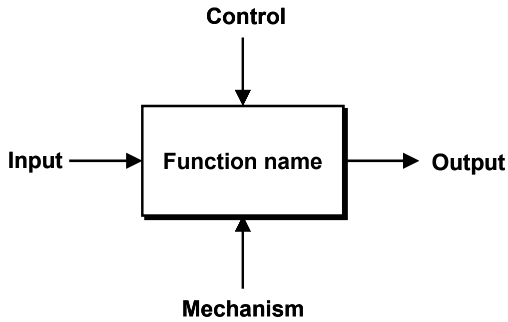


Figure 3-1: IDEF0 box.

The advantage of the SADT is that it gives a clear view of the whole process and of the activity sequencing. Thus it is a good support to evaluate the whole process but it also helps to optimize each activity by providing a better comprehension of its aim and its influence on the whole process. Indeed this type of representation makes it possible to work on one step without hiding the surrounding ones.

The disadvantages of the SADT representation are that it does not give information about the time and duration of the tasks; and such effects as the influence of the company organization are hard to represent. However we have decided to use the SADT as the organization is not the focus for the first step of this thesis, which is to analyze the current product development methodology in order to define the ideal one later. To complete the process analysis and define a process taking all the conditions of the real process into account, it may be necessary to make a second process representation with time and organization information and to confront it with the ideal process view.

The structure of this section will follow the top-down methodology of the SADT, starting with the representation of the global process down to the definition of elementary tasks. Not each activity will be detailed, only those of interest for the further definition of the modeling method.

3.2 Process Overview

3.2.1 A-0 Global Process

The process to be analyzed here is that of the stamping tool development. The top level of the SADT is shown in Figure 3-2.

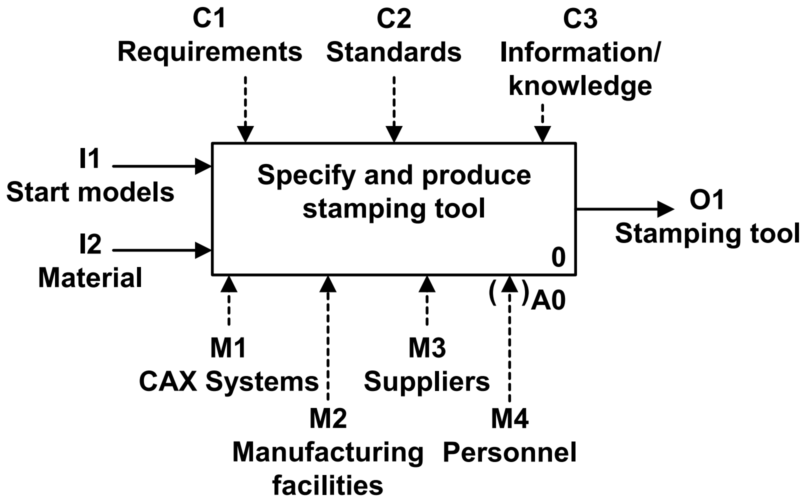


Figure 3-2: A-0 Global process.

As input for the stamping tool development, a start-off model is needed but it is not necessary for its shape to be accurate at this level since it has various areas of application. More details will follow in the next sections. In the same way, to simplify the representation, the requirements, standards, information, and CAx systems are not detailed here. The mechanism M4 is very important in this process but it will not appear in the following detail views as it would be redundant to display it in each diagram. In fact, this diagram gives a basic view of the overall process. The output of this activity is the stamping tool to be deployed in volume production.

Figure 3-3 depicts an overview of the structure of the process represented with the SADT method, with all the examined levels and their relationships.

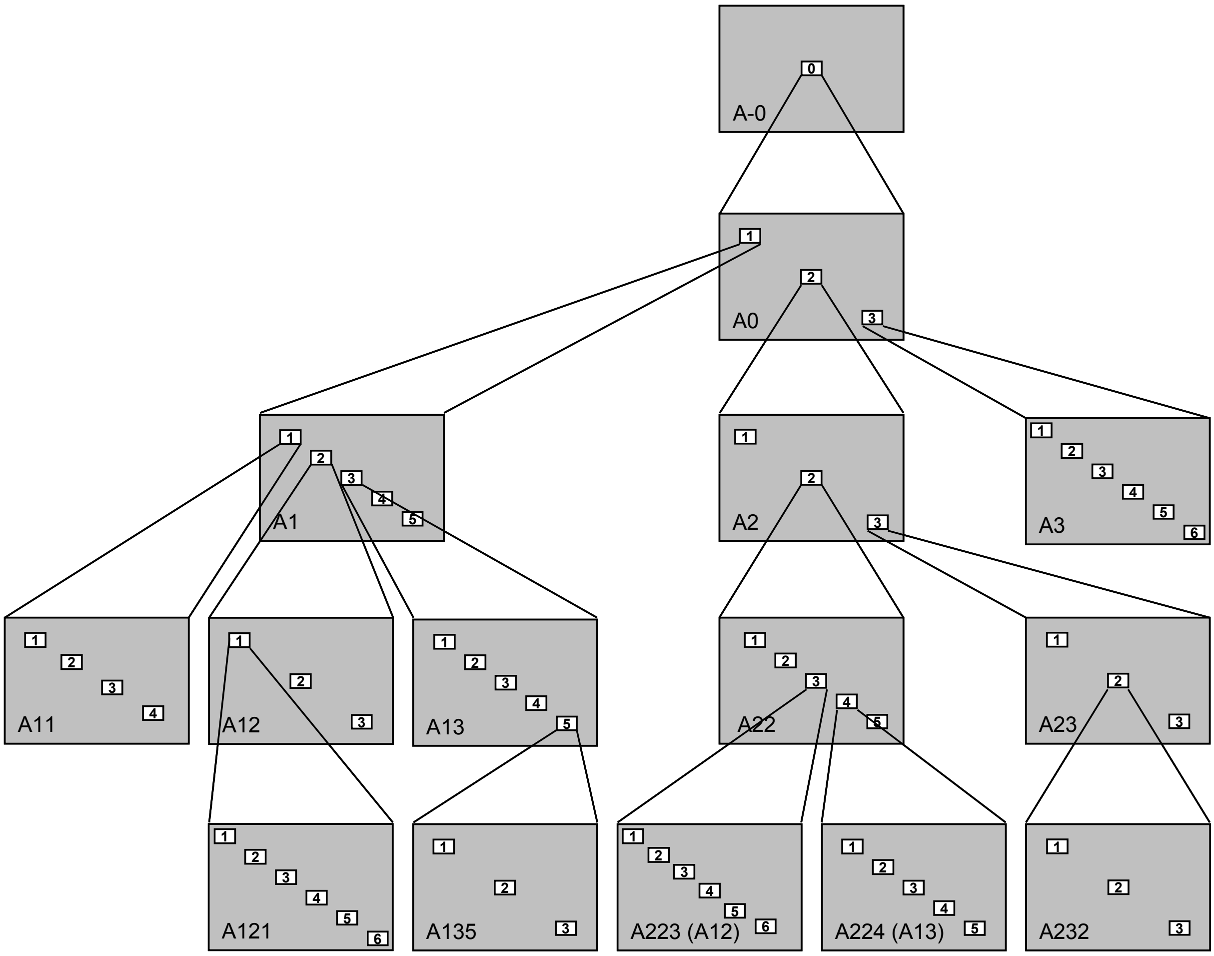


Figure 3-3: SADT overview.

3.2.2 A0 Specify and Produce Stamping Tool

The activity "Specify and produce stamping tool" has been subdivided into three activities as shown in Figure 3-4. We have included the development of the car-body component in the definition of this process as, during this stage, the initial surveys on the stamping tool feasibility are carried out, for example, the development of a prototype or the simulation of the stamping operation. Second, the interferences between the component and the tool development have the result that the optimization of the tool development process could only start with the optimization of the component development process. Similarly, the consideration of the tool production process is necessary to define the stamping tool development process. That is why it is also present in this representation.

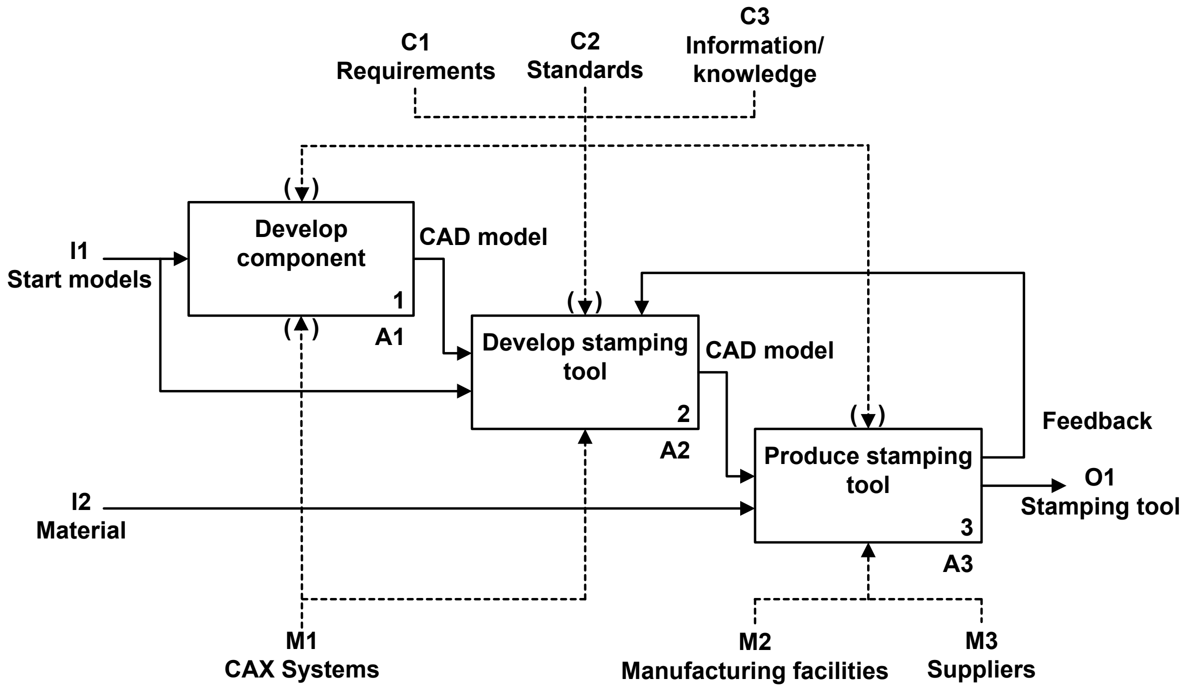


Figure 3-4: A0 Specify and produce stamping tool.

There are two key inputs for this process:

- I1 Start model: As the design process is based on CAD model, a start model is needed as input. This start model may be a prepared model containing a standard structure, or it may be an empty CAD-file, depending on the stage it is used
- I2 Material: The material is necessary to produce the stamping tool or its prototype.

The output of the process is the following:

- O1 Stamping tool: the aim of the whole process chain is to obtain at the end the stamping as hardware. Thus it is the main output of this SADT. The intermediary results are CAD models describing first the component geometry and second the tool geometry.

The control data are:

- C1 Requirements: In each step of the process, the requirements define the characteristics of the required results. They have to be met and will be used to evaluate and validate the results
- C2 Standards: The standards have to be considered by the designers at all time in order to generate correct data or compliant products.
- C3 Information/Knowledge: There is a lot of information used by designers to develop a product. This information could be accessible information such as previous CAD models, books of knowledge, intranet information pages but also implicit information with regard to designers' knowledge.

Finally the mechanisms of the process are the following:

- M1 CAx systems: Various CAx systems are needed to develop a product, including CAD, simulation, PDM or archival, and NC systems.
- M2 Manufacturing facilities: The manufacturing facilities are the set of machines used to produce the product. In this thesis, it is not necessary to define their nature as the focus lies on product development.
- M3 Suppliers: The suppliers provide the standard components needed for the stamping tool assembly.

3.3 A1 Develop Component

The level A1 of the SADT can be represented as shown in Figure 3-5.

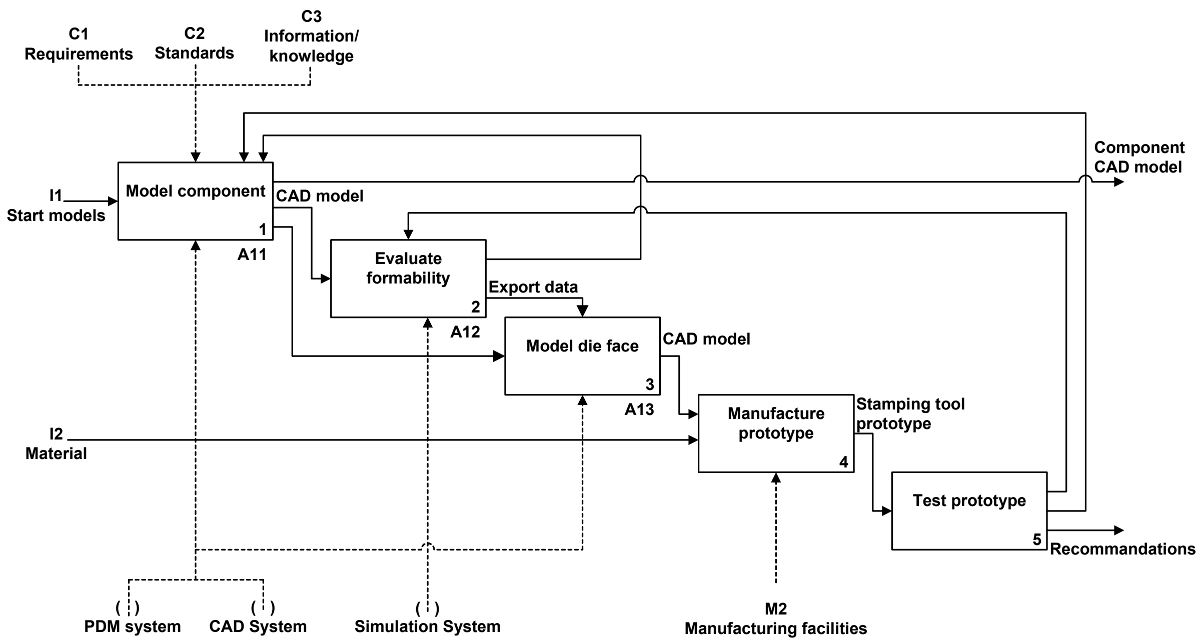


Figure 3-5: A1 Develop component.

This diagram stands for the development of the car-body component to be later produced by the stamping tool. In this representation the steps that are important for the tool design are represented; consequently, the adjacent activities of the component development are not displayed.

This part of the process begins with a start model that will be used to model the car body component. Since the introduction of CAD systems with history tree, the start model contains a prepared general structure that suits every component model. This structure is made first to standardize the models and make the representation of the component independently of the designer who done it, but also to optimize the modeling sequence. The aim is to make the modeling as efficient as possible during the model creation and during modifications.

When the first version of the component is achieved, the resulting CAD model is used to simulate the stamping operation (see details in section 3.3.2). The aim here is to evaluate the formability of the designed component. The simulation makes it possible to identify the problematic areas and to do recommendations to the designers who will change their models. The data are transferred from the CAD system to the simulation system but there is no geometrical data transfer in the reverse direction. That means that the modifications are made on the initial CAD model with consideration of the simulation feedback.

There is no special expectation about the structure and the format of the data delivered by the design to the simulation since the CAD data are converted into finite element (FE) data. But contents of these data and the design sequence have to be considered. Indeed, contrary to what is done today, it is not necessary to have a finished model to carry out the simulation. For an initial simulation, it is sufficient to have the main shape of the component. Otherwise, the component designer has to deliver a model that will be simulated and modified until it is correct.

The modifications required after the simulation may be of various kinds. They can be shape modifications as well as detail modifications. The things that often cause problems are the fillets and corners as they increase the constraints on the sheet metal during the stamping operation. Yet modifications may be more important if it is established that too many operations would be needed to produce the component or that the component could not be produced. Generally the importance of the modifications decreases with the number of iteration loops. Therefore, it would be interesting to identify the important problems before the component design is achieved.

Requirement 3-1
Facilitate modifications of the component after problem identification in the simulation phase.

After several improvement loops, the design is validated and sent to die face design. The die face design is based on a copy of the component model using the result of the simulation as reference. While the simulation software delivers geometry data in exchange format, the data quality is not good enough for the data to be used directly and it is also not possible to modify the data as the geometry is not parameterized. The aim of the die face modeling stage is to create a geometry that is as close as possible to the geometry coming from the simulation in order to achieve good quality data (parameterized CAD model). The problem is that the CAD die face may not overlay the simulated die face identically. Furthermore the modeling of this geometry is extremely time consuming in comparison to the time needed to create the geometry in the simulation software.

Requirement 3-2
Avoid remodeling die face in the CAD system.

Requirement 3-3
Make it possible to start the simulation and the die face design earlier.

The step following the die face design is the production of a stamping tool prototype. Based on the die face model, an NC program is generated and used to mill the face of steel blocks. There is no complex stamping tool design in this stage since it is made up of simple blocks that will be assembled on the press after the face milling. The inputs for this activity are the material and the die face model, and the mechanisms are the manufacturing facilities.

The production of the stamping tool prototype belongs to the component development since it is a full part of the component definition and optimization. The aim is to evaluate three criteria:

- The component formability and quality: It is necessary to verify if the component can be produced and to evaluate the quality of the component at hand. In this stage the thickness of the sheet metal is checked as well as the presence of folds and springback effects.
- The component functionality: The component is validated in view of its function in the car. To do so, it is assembled and various tests are carried out.
- The functionality of the stamping tool: The aim is to perform the initial evaluation of the stamping tool and to make recommendations to production tool design.

The simulation is able to identify most of the problems. This means that the prototype is foreseen to point out slight problems and to make the last optimizations. That is why there are loops from the activity "Test prototype" to "Model component" if problems are seen as coming from the component or to "Evaluate formability" if the problem derives from the die face. During the test phase, some modifications are made manually directly on the hardware, calling for validation in a new trial. The consequence is that the die face model does not correspond to the real tool anymore and that it has to be modified.

3.3.1 A11 Model Component

As mentioned, the component model has a standardized structure but its weak point is that it focuses on the component design and offers only few advantages for the following process. Only the holes have been considered as potential for the further process, namely the inspection planning. However the stage that directly follows component design is the modeling of the die face. The first draft of the component model is used to generate a die face in the simulation software and the final model is employed for the design of the die face. It is hence important to adapt the structure to the further model utilization in the die face design stage.

Requirement 3-4
Adapt the component structure to a further reuse in the die face design stage.

During the modeling of the component, designers take process information into account but nothing guarantees that this information will be considered exhaustively. That means that it is not possible to insure that designers have all the knowledge available about the next stage. This knowledge is important since the component that is being developed must observe some rules of formability and production cost optimization. If more information were available in the model, errors could be avoided and product quality improved.

Requirement 3-5
Integrate process information in the component model.

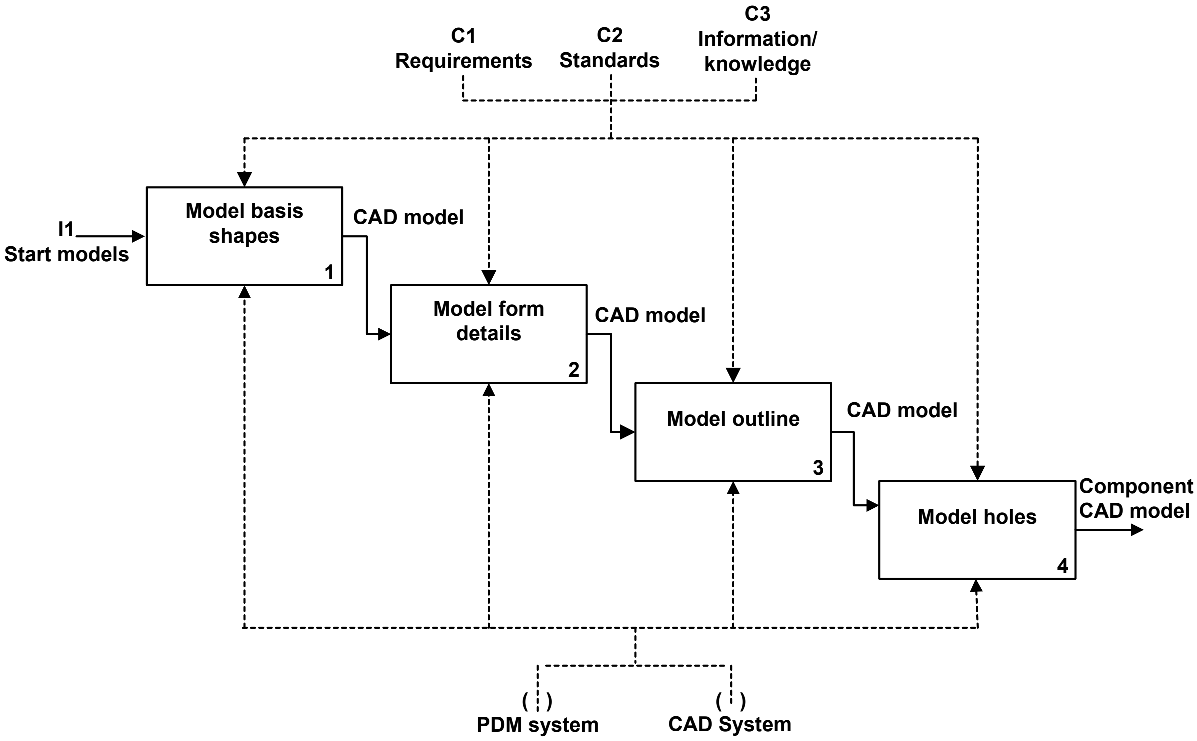


Figure 3-6: A11 Model component.

The modeling of the component may be divided into four activities as shown in Figure 3-6. The first step of the component modeling is typically the modeling of the main shapes of the component. Based on these, form details such as pockets, beads, or flanges are modeled. Then the outline is created, and finally the holes are trimmed all at once. This sequence may not be observed exactly, for example, if not all the information from the surrounding components is known. In this case, only a part of the main shapes with their corresponding details are modeled and the remainder is done later (see Figure 3-15).

For all these steps, a CAD application is used together with a PDM system for file archival. The control data are the design requirements and various information about the processes, for example, relevant standards or the previous model.

3.3.2 A12 Evaluate Formability

The evaluation of formability takes place several times in the overall process chain but we have represented it only once since the activities are invariably the same. The difference is that the input information does not have the same origin each time and that some control data are not always present. This activity is performed at different times during car-body component development to evaluate formability and offers the possibility to validate component design by predicting its behavior in the future stamping operation. It is also done in the stamping tool development phase to determine the final geometry of the die face.

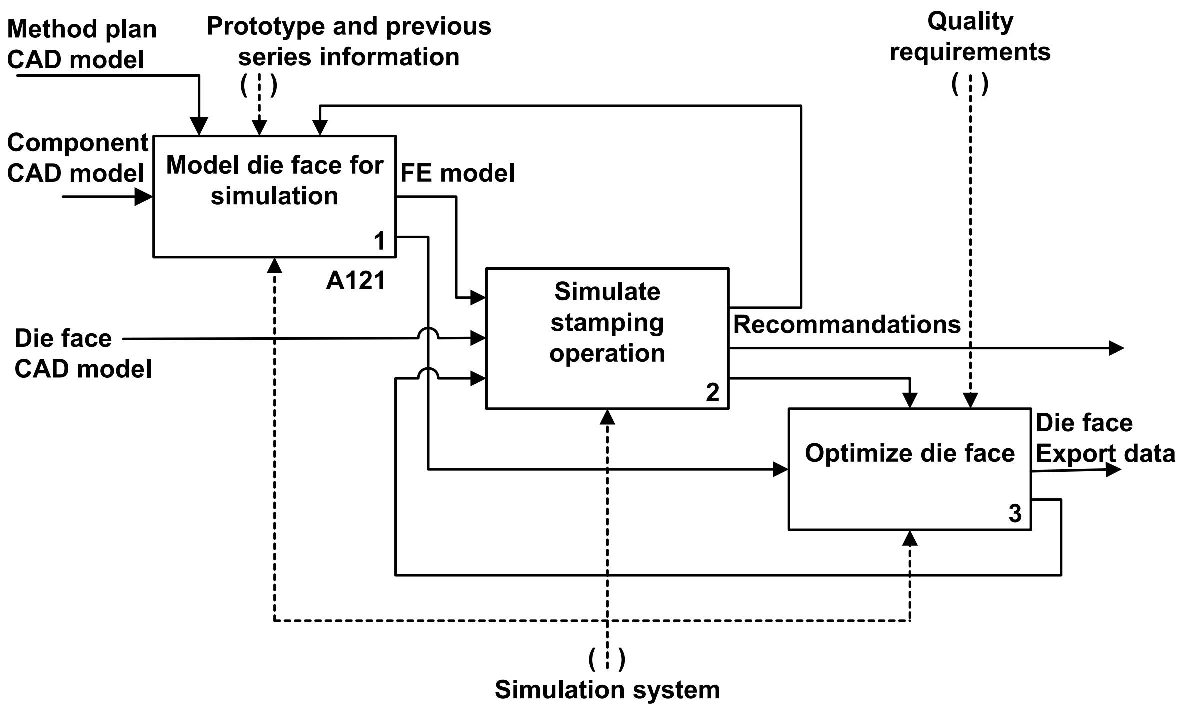


Figure 3-7: A12 Evaluate formability.

The SADT diagram corresponding to the activity "Evaluate formability" is depicted in Figure 3-7. The three activities contained in this diagram are carried out in the same system, which is, contrary to a CAD system, dedicated to a specific domain of application: the simulation of the stamping operation. It thus works very efficiently.

The first activity ("Model die face for simulation") receives data from car-body designers as input. Due to the iteration loops mentioned above, this data may be a first draft or a preliminary version. The aim is then to quickly generate the die face geometry to be simulated but not to model a quality geometry that could be used for further steps (milling). When this activity is carried out for the first time, only the data from the previous models are considered; yet if the simulation has already been done or if a prototype has been produced, the die face must be modeled, taking this information into consideration. Details on this task are set out in section 3.3.3.

The generation of the die face in the simulation system is based on FE geometry. This die face is used in the second activity called "Simulate stamping operation". In this stage, the formability of the component is evaluated and the die face geometry is validated. The input of the simulation is not necessarily the previously generated die face in the form of an FE model, a CAD model of the die face design may also be simulated to be validated a last time before starting the design of the stamping tool.

Regarding the result of the simulation, recommendations are made to one or the other step of the process chain. Modifications may be desired in the component geometry or in the die face design: this is represented by feedback arrows to these activities in the diagrams. If the simulation results are satisfying enough to avoid a modification of the component, the die face is optimized in the third activity ("Optimize die face"). Usually, there are several comings and goings between the simulation and the optimization until the quality criteria are validated.

3.3.3 A121 Model Die Face for Simulation

The detail of the die face modeling as it is done in the simulation system is represented in Figure 3-8. As all the activities of this stage are carried out in the simulation system, it is not possible to modify them. However, it is important to take up this process to compare it with another one that is almost similar: the modeling of the die face in the CAD system (see section 3.3.4). In fact both processes aim to model the die face but for different applications - the simulation or the stamping tool design. The question is to find out if this redundancy is necessary and if not, how it could be limited or avoided.

Requirement 3-6
Homogenize the die face modeling in the simulation and in the CAD software.

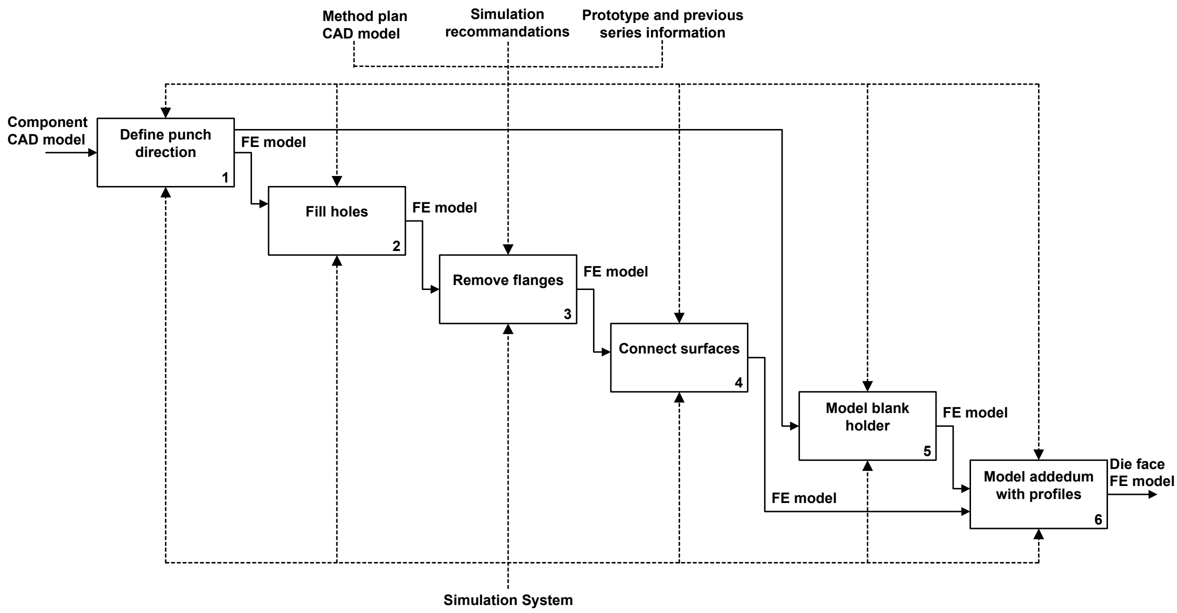


Figure 3-8: A121 Model die face for simulation.

The input data for this sequence is the component model as CAD file. It is converted into a finite element net by the system. The first activity that designers have to do is to determine the punch direction and the orientation of the component. If there are several components to be formed in the same tool, their relative position will also be defined in this step.

The component's holes are cut in a further operation so that the die face draws the component geometry without holes. Thus holes have to be inserted in the second step. Then the flanges that cannot be formed in the first stamping operation because of the relative angle to the punch direction have to be removed. Only the FE model is modified in these steps; the original CAD model remains intact.

In the case where there are two components, they have to be connected by means of a transition surface. Some areas present in the concave contour of the components also have to be designed. This is done in the fourth step called "Connect surfaces". As a result, designers obtain a unique surface without holes and with a smoothed contour.

The fifth activity consists of modeling the blankholder surface, which is a surface almost parallel to the component surface. This surface is adjusted using control profiles. Then both of the surfaces (component and blankholder) have to be connected using a surface called the addendum. Its role is very important for the stamping operation as it has a great influence on the finished sheet metal component quality. The addendum is modeled on the basis of profiles proposed by the system. The user has the possibility to manipulate each profile to obtain the result desired.

During the modeling of the die face, designers use information and knowledge about the previous volume-produced part and also feedback from the simulation or the prototype if it is not the first draft. If this stage takes place in production tool development, information from the method planning, where the overall production sequence of the component has been defined, is also available to designers. The die face is the final result of these six activities; it is subsequently employed in the same system for the simulation.

3.3.4 A13 Model Die Face

In this section, we focus on die face modeling, not for simulation but for the modeling of the stamping tool and the machining. The modeling of the die face is done for the prototype and for the production part stamping tool. As the requirements in both these stages differ, the same die face cannot be employed for both: first since the quality of the component and the performance and the material consumption have to be optimized in the die face modeling for the production tool and, second, since different downstream processes are employed. In fact, the prototype department deploys different technologies such as laser-cutting, which makes it possible to cut the holes and the flanges in each direction, whereas the cutting of a production part is subject to many more constraints. For this reason, the die faces do not look the same in the prototype and in development of the tool for volume production. Yet their modeling employs the same sequence in both cases as indicated in Figure 3-9.

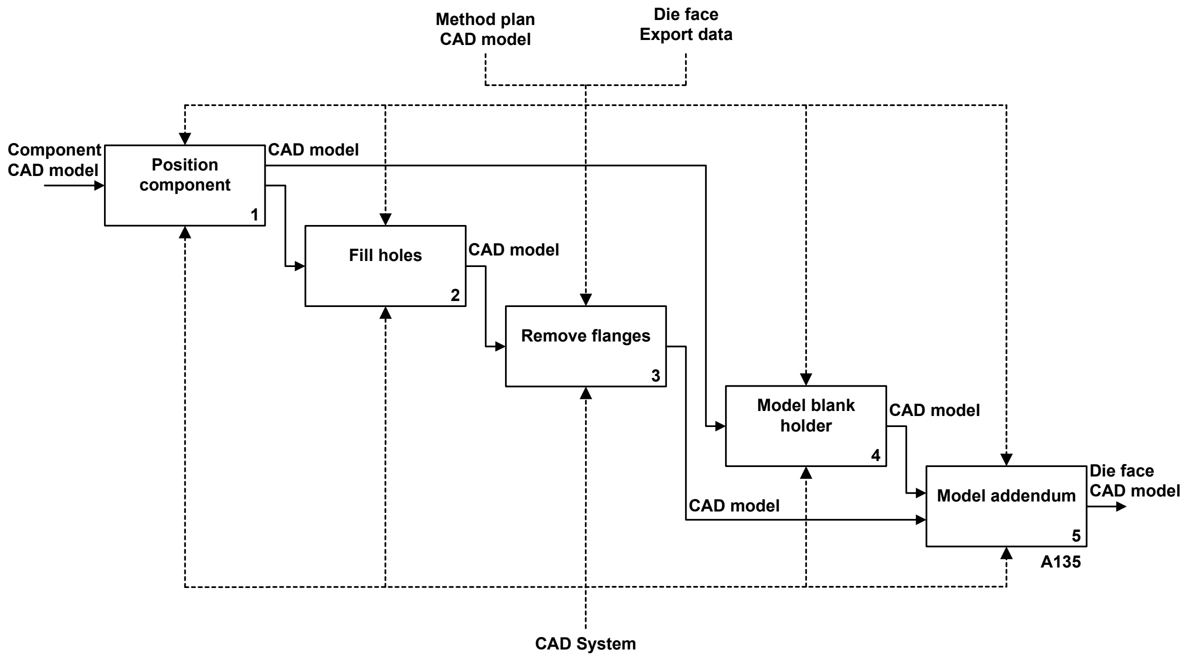


Figure 3-9: A13 Model die face.

The entire sequence is accomplished by die face designers in a CAD system. The input is the component model, but the result of the simulated die face is crucial as it is the reference to be remodeled. Therefore the component model and the die face model from the simulation system are inserted in the file dedicated to the final die face modeling.

The first task is to position the component relating to the punch direction as it was done in the simulation system. The transformation operations (translation and rotation) of the CAD system are used. Then the holes have to be closed. This cannot be done in one click as in the simulation system, which utilizes a function to fill all the holes in one operation. Instead, each hole edge has to be selected and filled.

In the third activity, the aim is to remove the unnecessary flanges. This can be done by deleting patches or, depending on the CAD system, by trimming the surface with extracted edges. The activity "Model blankholder" is almost the same as the one in the simulation system except for the fact that designers do not automatically get a suggestion from the system, instead having to model it completely. Both results of activities 3 and 4 are then used in the fifth activity called "Model addendum". This is detailed in section 3.3.5.

The analysis of this sequence shows that there are steps that could be avoided by anticipating the die face modeling in the component-modeling phase. For example, the holes or the flanges are modeled in the component and filled or deleted in the die face. The question is whether these operations could be avoided.

Requirement 3-7
Facilitate the modeling of the die face on the basis of the component model.

3.3.5 A135 Model Addendum

The details of the addendum modeling sequence are given in the diagram of Figure 3-10. It is the last step of the die face modeling and is done in the same file but had to be shown in a separate diagram to simplify the representation.

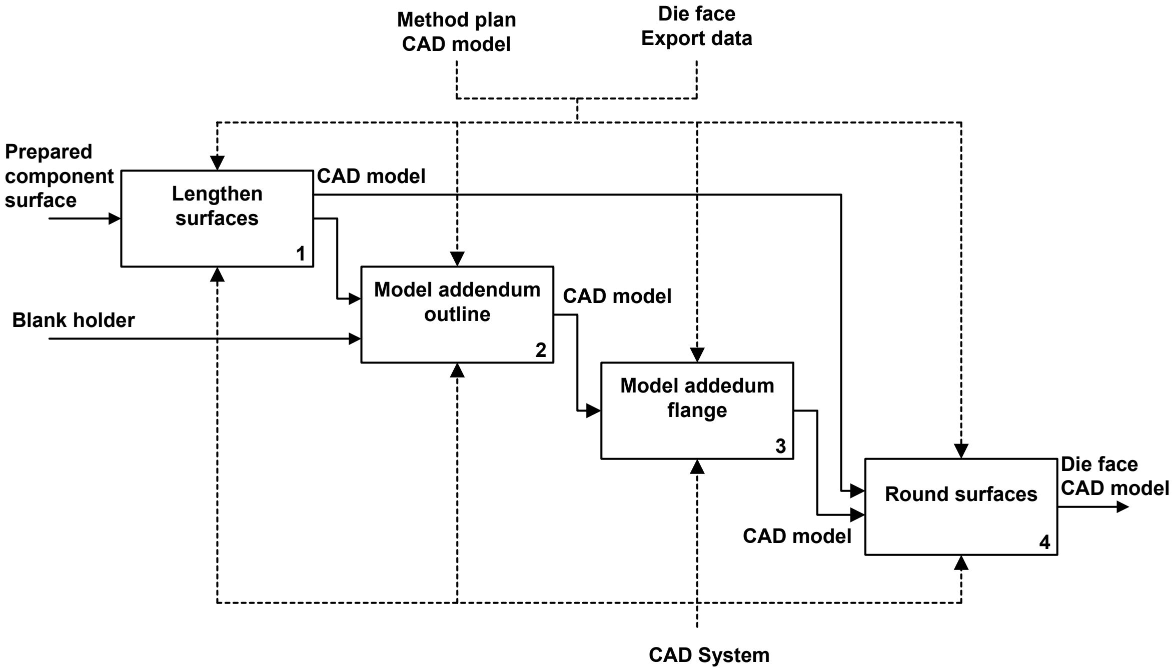


Figure 3-10: A135 Model addendum.

The component surface, without its holes and part of its flanges, is lengthened in the first activity to obtain a basic surface that is rounded later with the addendum flange. To lengthen the surface, its contour has to be extracted and extrapolated. This is sometimes problematical: the curvature of the surface and its contour can lead to twisted surfaces during the extrapolation. If there are two components to be formed by the same stamping tool, they have to be connected (as in the simulation system) by a surface tangential to them. This operation is also not as simple as in the simulation system.

The addendum outline is modeled in the second activity, which aims at creating the guiding curve for the addendum flange. This curve has to follow the component contour and to be located on the blankholder surface. The subsequent activity targets modeling the addendum flange on the basis of the previously modeled outline. Finally the lengthened component surface, the addendum flange, and the blankholder surface have to be rounded to obtain the die face surface.

The modeling of the die face can be a time-consuming task if the component model is complex. Additionally, it is difficult to reproduce exactly the same geometry as that created in the simulation system. Thus there is potential to improve this stage (several requirements have already been cited above).

3.4 A2 Develop Stamping Tool

Figure 3-11 depicts level A2 of the SADT, "Develop stamping tool". The activity regroups the set of activities used to specify the production part stamping tool. First is the elaboration of the production facilities plan. While this is not directly concerned by the CAx process chain, it is essential for the definition of the stamping tool since it establishes which press the stamping tool has to be designed for, thus providing information about the dimensions of the tool and its working principle (simple or double-action tool). The production facility plan is typically then set up on the basis of the press capabilities in the die shop so that the component characteristics do not play an important role for the choice of the press.

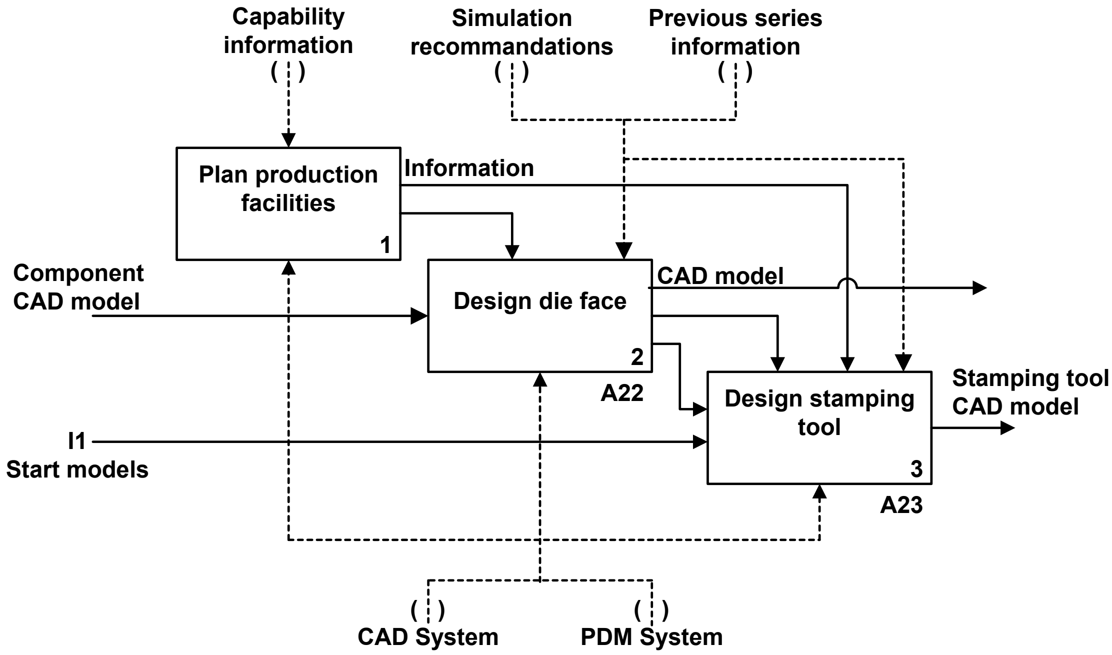


Figure 3-11: A2 Develop stamping tool.

The second and third activities make up the real development of the stamping tool. The die face design is done on the basis of the component model with the information from the production facilities planning; the result of the die face is employed for the stamping tool design. For this, a start-off model is also used as input as well as information from the previous model as control data. There may be iteration loops within each activity but there is no loop from one activity to one other in this process. The final result is the stamping tool model that is used to produce the physical product.

The first activity is not described in detail as only its result is important for this work. However the activities "Design die face" and "Design stamping tool" are elaborated in sections 3.4.1 and 3.4.2, respectively.

3.4.1 A22 Design Die Face

The design of the die face takes place a first time in the product development to create a prototype. In fact it is done in the simulation system. In production part tool development, the die face design process is a more complex process where simulation is only one of the steps. The first task of this procedure is to begin the modeling of the die face (Figure 3-12). However, this does not deal with a full modeling of the die face but rather with the drawing up of a first draft that could be used as input for the method plan definition and for the stamping tool modeling. The die face can be begun before the method plan is at hand thanks to the information from previous model and from the prototype and simulation tests.

In the second activity, "Define method plan", the sequencing of the operations necessary to produce the component is defined. Not only the characteristics of the stamping operations but also the followings ones (flange folding, hole trimming) are specified as each operation has influence on the others. However, in this work, we focus only on the results of the method plan definition that concern the deep drawing aspect. Important aspects of quality and cost requirements have to be considered in this stage since they determine the way the component is to be produced.

The method plan is generated in the CAD system on the basis of the first draft of the die face. It deals with the modeling of information for the further step. For the stamping tool operation, this information is in the form of curves describing the main characteristics of the stamping tool: the punch direction, the punch opening, the draw bead's guiding curve, the theoretical blank contour, the flanges to be developed, etc.

The third activity, "Evaluate formability", is repeated. In the beginning, the model of the first activity is used, but in the next loops a more refined model from the fourth activity is employed. In fact, the third activity is identical to activity A12, which has been described in section 3.3.2 for the evaluation of the prototype die face. The difference with the formability evaluation for the prototype is that, in production tool development, the die face is not modeled in the simulation system but in the CAD system. Hence, the first activity in diagram A12 is ignored.

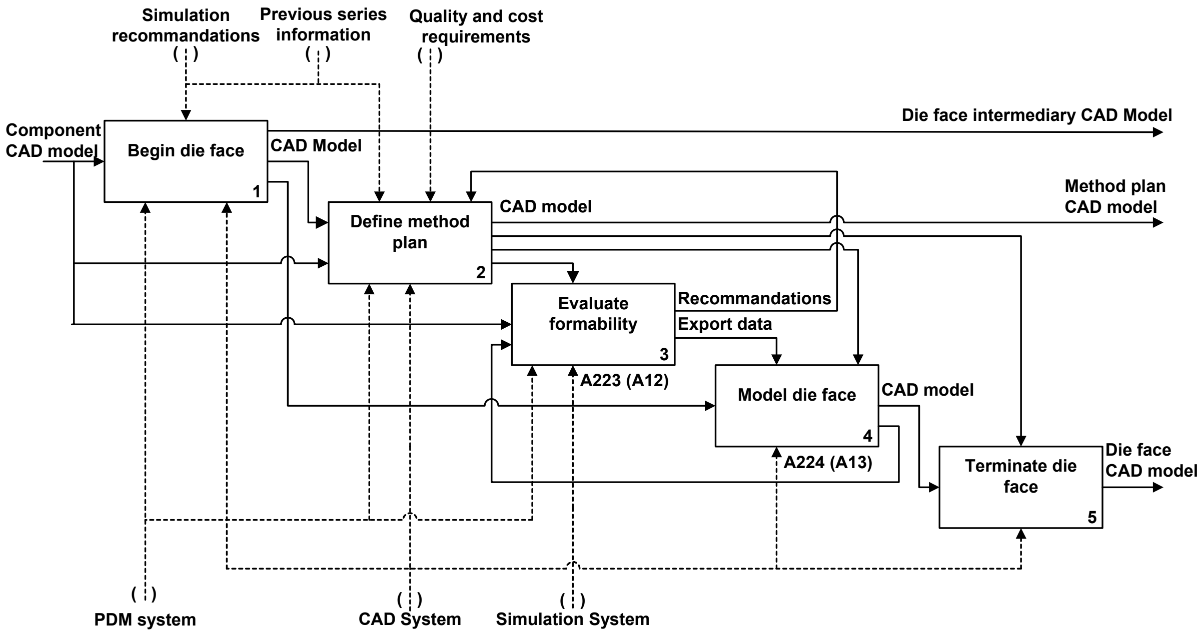


Figure 3-12: A22 Design die face.

The fourth activity is also similar to one already described: A13. It consists of modeling an achieved version of the die face in the CAD system. There are several improvement loops between activities 2, 3, and 4 until the die face is optimized.

Before delivering the die face to the stamping tool modeling, the die face has to be finished in the fifth activity. The aim of this task is to check the model quality and to generate all the necessary data. In fact, the die face is the basis for other data, namely the punch face, the blankholder, and the matrix face. It is also necessary to deliver an offset surface of the die face to later model the thickness of the various stamping tool components.

Note 3-1
Depending on the organization of the company, distribution of the tasks may vary: for instance, the definition of the method plan and the evaluation of the formability may be done by the same designer and the modeling of the die face by a second one, but it is also possible that all these tasks are carried out by the same person.

3.4.2 A23 Design Stamping Tool

In this section, the stamping tool design process is described. The level A23 of the SADT is presented in Figure 3-13. There are three main stages in this process. First, a flow chart has to be drawn to represent the trajectory of the sheet metal trough the press line. In particular, it defines the position and the orientation of the sheet metal in each station. This information is important for stamping tool design to know the position of the die face in the press.

The input for the creation of the flow chart is an empty start model. The flow chart is done using the result of the method plan and the die face as information as well as the characteristics of the chosen press line. As a result it yields a 2D drawing created with help of a CAD system. It represents a side view of the press line with the trajectory of the sheet metal center-point as curves.

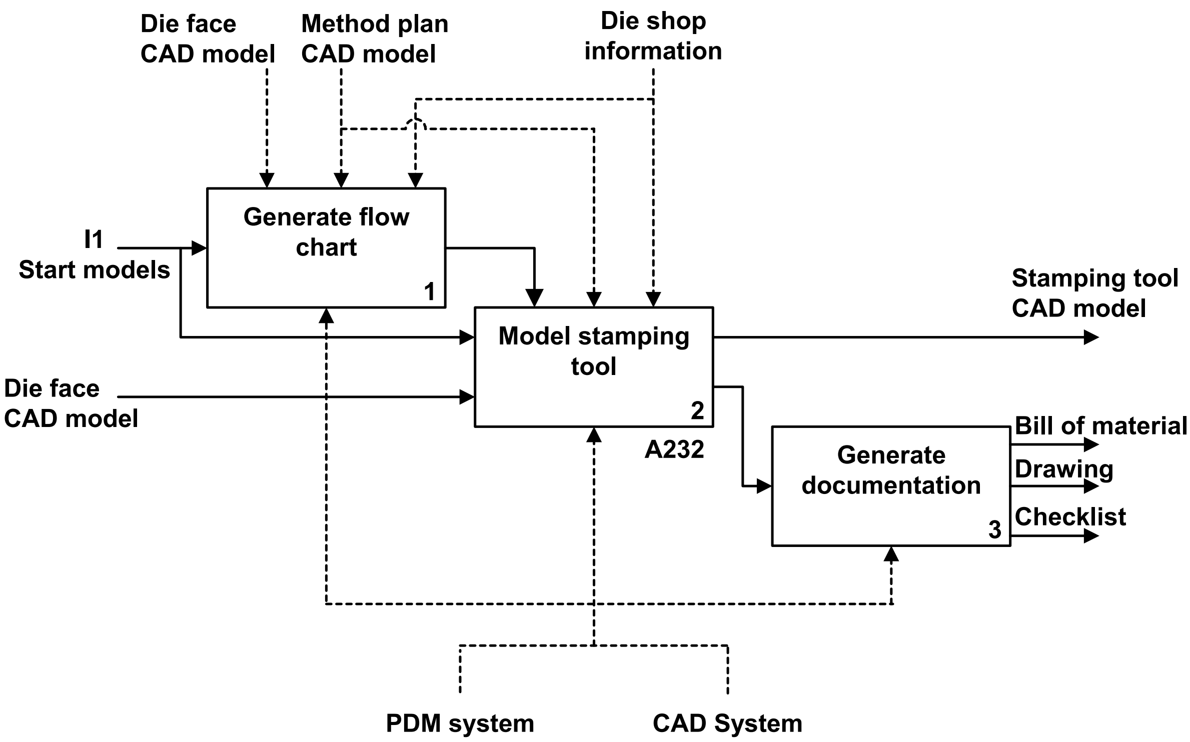


Figure 3-13: A23 Design stamping tool.

Typically, the drawing of the flow chart is printed by designers so that it can be consulted during the stamping tool modeling stage. This means that it is not an input for the next activity but rather control data. Thus the activity "Model stamping tool" begins with a start-off model and the die face model as input. In addition to the flow chart, it receives information from the die shop as control data. All the permanent control data such as the standards and the requirements are also important but they are not shown in the diagram for reasons of enhanced legibility. The details of this activity are presented in section 3.4.3. Its result is the stamping tool as CAD model; it is then used as the basis for the generation of the documentation for the following process as explained in the next section.

The first necessary document is the bill of materials, which is used later for production and the purchase of the components. The bill of materials is normally generated using a function of the CAD system. A deficit is that much information has to be inserted later in the bill of materials although this could have been done in the CAD system.

Requirement 3-8
Evaluate the potential of automation in the generation of the bill of materials with additional information.

Apart from the bill of materials, designers deliver drawings of the stamping tool to the production department, which needs them to plan the production and montage of the stamping tool. To close and validate the design phase, a checklist has to be completed. It contains all the points that have to be verified to assure the conformity of the stamping tool model: the number of given elements, the conformity to the standards, the fitting of the press, etc. That means that the design is inspected afterwards and that errors have to be corrected.

Requirement 3-9
Avoid errors in the design phase instead of correcting them.

3.4.3 A232 Model Stamping Tool

The difficulty in utilizing the process of the stamping tool modeling is that the currently employed method is largely not standardized and it is hard to define a unique modeling sequence. That is why fewer details are given in this section than in the section describing die face design although a stamping tool is composed of many components. A simple-action stamping tool, for example, comprises four main parts (upper die, lower die, punch and blankholder) and many additional standard components.

Normally, it is possible to begin the modeling of the stamping tool as soon as the die face dimensions (Activity 1 of the diagram A22) and the chosen production press (Activity 1 of the diagram A2) are known. In this situation, where merely information is given but no CAD data are available, the only part of the stamping tool that can be commenced is the lower die, as it requires knowledge of the chief dimensions of the die face but not of its geometry. The other components are modeled when there is more information available. Thus the modeling sequence depends slightly on the status of the information, but the determining factor is the designer, who is free to model the components in any desired order. Moreover, two designers may work on the same stamping tool model, with one making the lower part and the other the upper parts.

The diagram in Figure 3-14 hence represents a general process that is always valid independently of each component modeling. The tools employed to perform this process are a PDM system and a 3D CAD system.

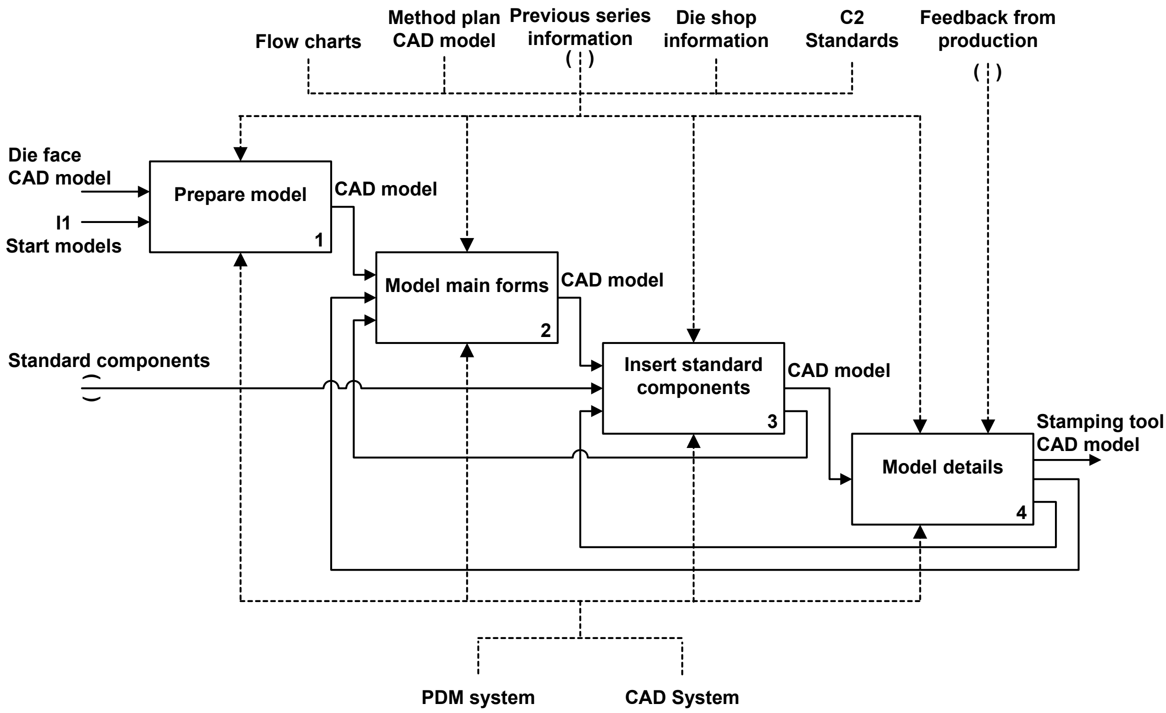


Figure 3-14: A232 Model stamping tool.

The start model used to begin the modeling of the stamping tool is an assembly model containing a standard structure. It is prepared so that the designer knows where each component should be placed; yet the model structure is not specified further in depth so that each designer has the choice to structure the component models desired. In the first activity, designers insert the die face in the start model and consult the information in the flow chart to create a reference system relating to the die face center-point. Sometimes it is also necessary to correct the die face because of quality problems. In fact, if the die face surface contains laps and tangential discontinuities it has to be rectified since sending back the model to die face designers would take too long. After that, the designers schematically model the main characteristic of the press, above all to visualize the position of the press table and its dimensions. That means that the geometry of the press has no other function than helping designers in the modeling stage.

Whatever the component to model is, the modeling sequence generally starts with the main shapes and goes on with the details. However it does not mean that the details are modeled only when all the main shapes are achieved: it is possible to create the main shapes of an area of a component and to insert its relating details and then to create the main shapes and the details of another area. These two possibilities are shown in Figure 3-15.

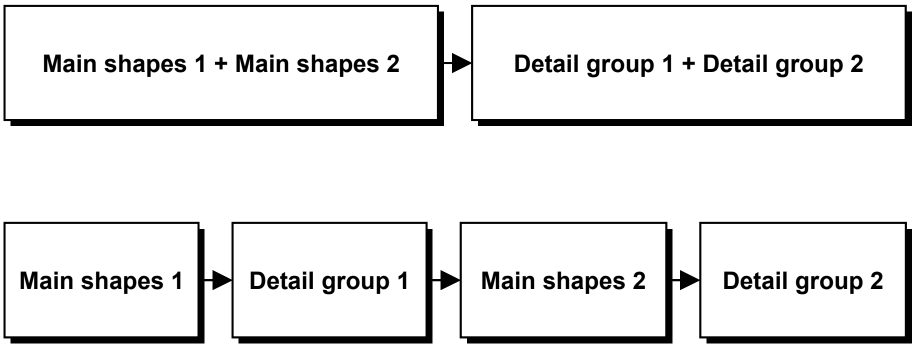


Figure 3-15: Two possible modeling sequences.

In the SADT diagram A232, the fact that there can be coming and going between the main shape modeling and the detail modeling is represented by the loops between activities two, three, and four.

The standard component insertion (Activity 3) is placed between the main shape modeling and the details modeling since they represent whole components whereas details are only parts of a component. Thus, in the order of importance, they may be considered before, yet this is not mandatory.

When the model is finished, it is delivered to the production department, which may give feedback so that slight modifications are required. The aim of the modifications is mostly to optimize the machining time or to correct situations where an area cannot be accessed for the machining. Modifications are treated in the activity of the detail design.

3.5 A3 Produce Stamping Tool

Figure 3-16 portrays the sequence of stamping tool production. It is interesting to analyze this process as well since it is the recipient of the data generated in the previous processes. Thus the analysis of the production process can help to identify important requirements for the design phase.

The inputs for this process are the CAD models of the die face and of the stamping tool. In principle it would be possible to be content with the stamping tool model because its geometry contains the die face. But the reason why the die face is also delivered is that it could have undergone slight modifications that were not reflected on the stamping tool so that this one is not up-to-date. By using the die face model directly for the machining, it is guaranteed that this information is correct.

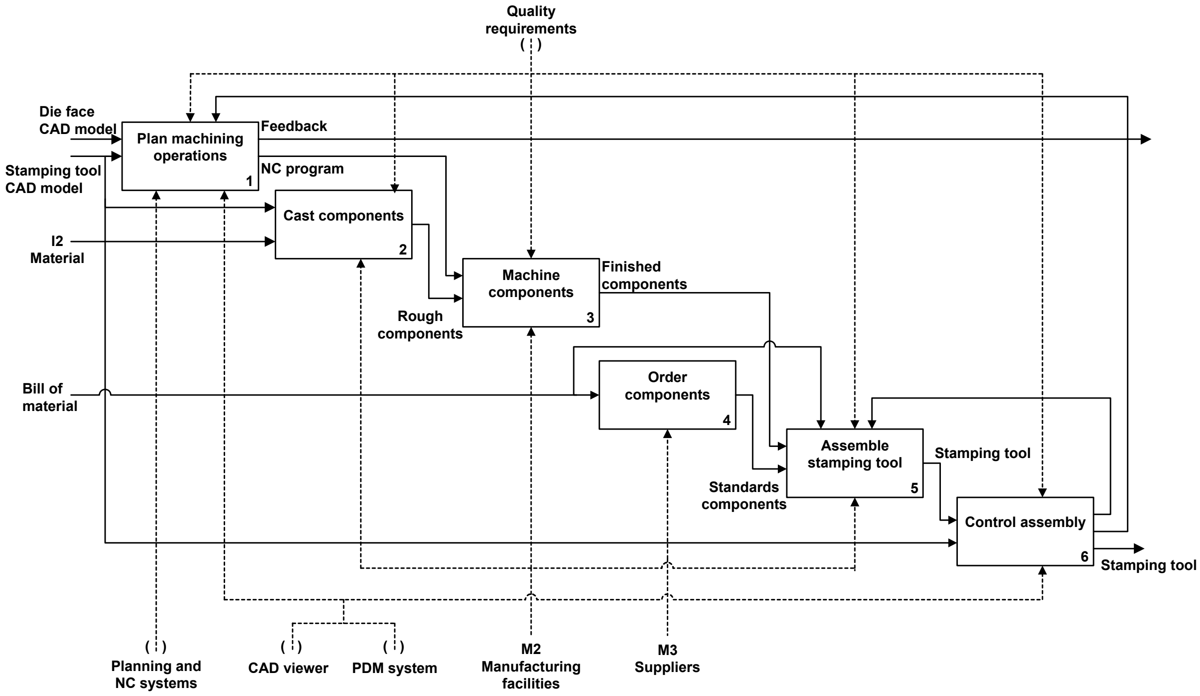


Figure 3-16: A3 Produce stamping tool.

Requirement 3-10
Clarify the problem of data updatedness, particularly for the die face data.

The aim of the first activity is to generate the NC program necessary for the component machining. It is done in three stages: sequence planning generation, activity network generation and NC program generation. The sequence planning is carried out on the basis of the bill of materials and the stamping tool model. It resembles a table with general information such as the component references, their main dimensions, their positions, and their materials. Then, the following information is defined for each machining operation:

- Designation of the operation
- Number of axes of the machine employed
- Dimensions for the holes
- Machining quality
- Number of items
- Estimated machining time
- Notes

With this document, the engineers register all the machining operations to be performed and estimate the overall machining time. Additionally, this is an opportunity to provide feedback to stamping tool designers if some slight modifications that would simplify the machining sequence in the sense of processing time should be required.

The definition of the sequence planning is time consuming and may take up to several months. The system used to do the sequence planning offers a support for the representation but does not create a direct and associative link with the CAD model although the same information is present in both documents under different forms. That is why it would be useful to directly associate the CAD model and the sequence planning.

Requirement 3-11
Make an associative link between the CAD model and the sequence planning.

To identify the surfaces that have to be machined, they have to be colored according to a given color convention defining the desired surface quality. This is useful for the visualization of the surfaces to be machined but is not process secure in the sense that surfaces may be omitted by the production engineer. In addition, it fails to provide all the necessary information (for example, name, function of the surface).

Requirement 3-12
Insure the identification of the surface to be machined.

Once the sequence planning and all the design modifications have been carried out, a second document has to be created by production engineers. This is the activity network, which defines the chronological order of the machining sequence. It looks like a diagram representing the operations as rectangles and connection lines to describe the paths between them.

The sequence planning is made with the stamping tool model and the sequence planning as input. Two important criteria for the definition of the machining sequence are the position of the areas to be machined and the milling cutter to be deployed, so that the areas having the same properties are machined in succession. It this step, it would again be interesting to prepare the stamping tool model in order to facilitate the creation of the activity network.

Requirement 3-13
Consider information about the activity network in the stamping tool model.

The sequence planning and the activity network are then sent to the die shop where the NC program will be compiled and run. In parallel to the creation of the definition of the activity network and the NC program, the main components of the stamping tool have to be cast (Activity 2). To do so, a foundry is mandated and receives the stamping tool CAD model. Its first task is to make a polystyrene model in real size to be used later as the core for the component's casting. The difficulty in the building of the polystyrene model is that it does not exactly match the CAD model. In fact the CAD model gives a representation of the finished component whereas the polystyrene model corresponds to an intermediary stand of the component: it has the shape of the cast component before having been machined.

In the cast component, there are no holes and the surfaces to be rectified are offset to offer the necessary thickness for the machining. As a consequence, the foundry has to adapt the CAD model to create the polystyrene model. There is hence a risk of error and casting of a non-compliant component. For this reason, the polystyrene model is checked by the stamping tool designer before casting. If faults are found, the polystyrene model can be corrected once it is built by removing or gluing material, which is less expensive than modifying the stamping tool itself. The issue of the polystyrene model creation thus has to be tackled to insure the validity of its geometry.

Requirement 3-14
Adapt the stamping tool CAD model to the polystyrene model's needs.

The third activity is performed when the NC program is finished and the cast components have been delivered. It consists of machining the surfaces to be rectified as well as the holes. In particular, the die face is machined in this step. According to the bill of materials, the standard components are ordered from the suppliers (Activity 4). And once all the parts are available (cast components and standard components), the stamping tool is assembled (Activity 5) and quality controlled (Activity 6). The control phase serves to identify the problems and, depending on their importance, there is a feedback loop to the NC programming or to the stamping tool mounting.

3.6 Conclusion

The first chapter of this part has provided an overview on the current surveys regarding the development process. It has shown the possible solutions in this domain, but it has also demonstrated that there are gaps and improvement potentials in some areas such as the domain of templates and design methods taking process requirements into account.

On the other hand, the solutions presented in the scientific world are not necessarily transferred to the industry and there is a significant delay between the achievement of research and the transfer of results into practice. In many cases, not enough attention is paid to limitations imposed by the practical implementation; that is why it was also necessary to record the current stamping tool development process. The process analysis has given a practical overview of this process. Although this process has been analyzed in only one company in the branch, it has been depicted as generally as possible so that this company's particularities are not detectable.

In each step of the process analysis, requirements have been listed and will be used in the third part of the thesis to improve the situation in the stamping tool development process. It is recognizable in the entire process that most of the activities are carried out sequentially and that the information exchange is not really dynamic. That means that *most of the time information is delivered when it is completely defined*. In each step of the process chain, little information about adjacent steps is available. Thus, each actor concentrates on its own task and does not have a global view of the entire process: the available design methodologies lack a holistic view, aiming solely at optimizing the activity considered.

Finally *there are redundant tasks* being repeated several times at different places of the process chain as well as tasks that do not have a clear utility and are achieved to compensate for imperfection of the process. In addition, *there are routine tasks* that have to be accomplished in the same way each time a new design is started, without providing any improvement for the product.

PART II
Concept

Chapter 4
Functional Elements

4.1 Introduction

This chapter has for aim to identify relevant information of the designers' knowledge for the product design process and define the way it can be integrated in the design task. The first section is intended to define the origin of the problem and in the next ones a solution based on "functional elements" is introduced. This contribution makes it possible to leave the classical geometry-based modeling methods and can be classed in the field of knowledge based modeling methods.

4.2 Formulation of the Problem

Before examining the overall process chain and the advantages that an integrated model can bring, it is important to take a look at the design itself and to define what the aim of the design stage is and how it works.

Definition 4-1: Design
Design is the process of developing and representing a product's form and function that serve its intended purpose.

The task of designing uses the requirements established to answer the demands of the customer or the person who has mandated the work as input. Thus designers have a list of expectations that they have to meet. In other words, they have to define a product that yields a technical answer to characteristics that the customer expects. Each pair "expected characteristic/technical answer" can be seen as a function.

Definition 4-2: Function
The function of an element is its role in a systemic aggregate.

On the basis of their experience, their know-how, and other information sources, designers imagine a product that is the sum of all the functions that it has to fulfill and then represent it in a CAD model. Between the idea of the design and the final result of the design as CAD model, there is a step where designers have to think over how they could represent the geometry of the product with the available CAD system functionalities.

Therefore, there are two geometrical images of the product: the image that designers have in their minds and the image in the CAD system. The transition between both images is not actually done directly. *The intermediate step consists of decomposing the geometry of the product into elementary geometrical elements that typically do not make any engineering sense if they are seen independently from each other.* This phase where designers have to define how the product could be represented is the modeling methodology. This methodology is extremely dependent on the working of the CAD system. Most of the CAD systems employ 2D profiles that are used to create elementary solids (for example, by extrusion), which are then combined using Boolean operations.

To illustrate this method, let us look at a simple component composed of a slab with a slot hole and consider two aspects: "What does the designer wish to represent?" "How will the component be represented in the CAD system?"

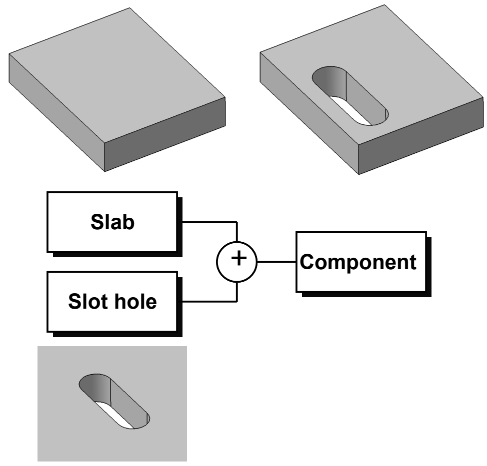


Figure 4-1: Decomposition of a component into functions.

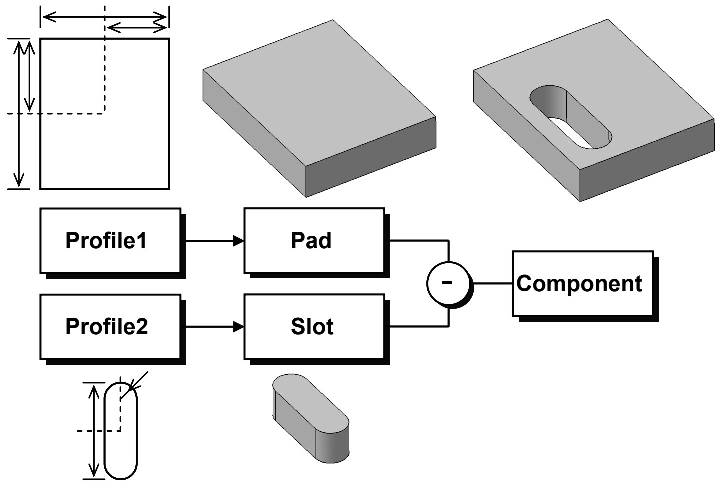


Figure 4-2: Decomposition of a component into modeling elements.

Figure 4-1 illustrates how the designer considers the component and Figure 4-2 illustrates what has to be done to achieve this result. Hence, the designer analyzes an idea and establishes that the slab is a rectangular pad and that a profile needs to be drawn to generate this pad. In the same way, the slot hole is created. In the present case, this reflection is not very complicated, but it shows the problematic of the modeling methodology.

The use of profiles or sketches forces designers to create two-dimensional elements whereas they wish to generate three-dimensional ones. Moreover, the parameterization of the sketch might also be different from an intuitive parameterization if the designer does this task as quickly as possible. The geometrical parameterization of an element has two aims: the positioning of the element and the definition of its dimensions. If particular attention is not paid to the parameterization, this may result in no clear separation between positioning and dimensioning and the constraints used to define a sketch often have both functions.

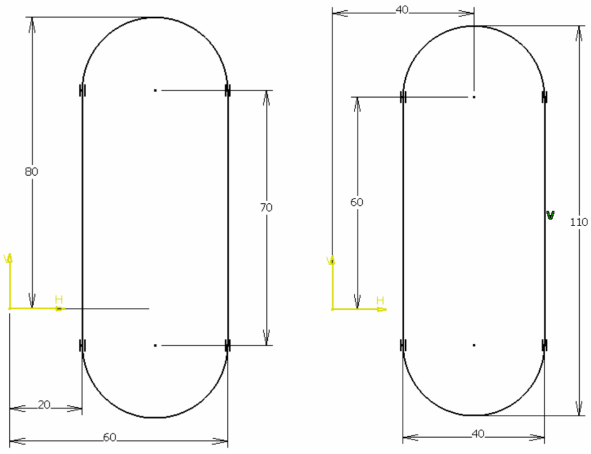


Figure 4-3: Two examples of profile parameterizations.

Figure 4-3 depicts two ways of parameterization. In the first example, the length and the width are not directly drawn. The width is given by the difference between the distances from two segments of the slot to a reference system. The length is given by the distance between both circle centers added to the diameter that is equal to the width. In the second example, the main dimensions (width, length) are given and the positioning of the slot is done using one vertical constraint and the position of one circle center.

In fact, a parameterization is not good or bad. All variants can be acceptable if they are justified. What would be important at this point would be to know why the designer needs this slot hole and to deduce how it should be parameterized. Many other options are available to define this sketch. For example, if the slot hole is used as a guiding notch for a pin whose diameter is equal to the width of the slot, a clever dimensioning would be to define the slot with the circle diameter and the length between both circle centers since this corresponds to the distance that the pin can cover. For the positioning, we can use a reference element for the orientation and coordinates of the center-point of the slot or we could use distances between the slot edges and the slab edges, for instance.

The problem of how an element can be parameterized will be discussed later. However, what is important to note here is that the parameterization has to be justified by the element's function and the use of sketches is a detour that should be avoided.

If the CAD system worked in a way similar to the designer's way of thinking, three-dimensional objects could be created directly in the system with parameters that make sense for designers. The modeling sequence would be the following:

- Create a slab with given length, width, and thickness (without using a sketch)
- Insert a slot hole with given position, orientation, width, and length

In summary, we can say that a CAD model is a means to virtually represent the idea of the product that designers have imagined and that the modeling step starts them off thinking about functions. Figure 4-4 illustrates this remark.

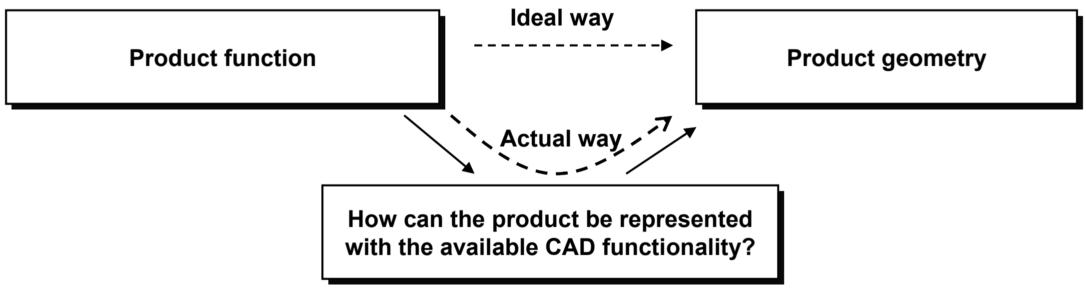


Figure 4-4: Function oriented modeling way.

To avoid the intermediary step, it is necessary to define new modeling methodologies that correspond to the engineers' way of thinking.

4.3 Functional Element

4.3.1 Introduction

As explained in the previous sections, it would be useful to provide designers with modeling methodologies that take into account the way they work and how they conceive a component. Designers think of elements that fulfill functions. We therefore talk about functional elements.

The first requirement set for a functional element would be to enable users to work with objects whose behavior (see definition in section 4.4.1) is the same in the CAD system as in their mind. However, a functional element can provide many more advantages than the simple description of the geometry, particularly with respect to knowledge integration. The following section will define all the requirements of a functional element and explain how it can carry knowledge. Next, various types of functional elements will be introduced.

4.3.2 Characteristics of a Functional Element

Definition 4-3: Functional element
A functional element is an engineering element that fulfills a mechanical function in the real product and is represented in the virtual design system as an identifiable object with process and knowledge information.

Engineering object
As the definition states, a functional element is, first of all, an engineering object. The aim is to define functional elements in a virtual model as closely as possible to their images in the designer's mind or in the real world.

Concretely a functional element can be a part of component, a whole component, or an assembly. Each of these items has its own function and characteristics depending on the context. For example, a car is an assembly whose function is to carry people from one place to another. If we look at it in more detail, the car is composed of sub-assemblies that also have functions, for example, the engine or the suspension. In the same way, each component (screw, piston, etc.) has a function. We can continue this breakdown into elementary items such as holes or beads.

It is important to note that the subdivision stops when the corresponding element does not have a function in sense of engineering. That is to say, elements in a CAD system such as sketches, lines, vertexes, or faces will not be considered here as functional elements.

Identifiable objects
Today's CAD systems provide a representation of the part structure, called the specification, structure or history tree. In this representation, each element has its own identification attributes in the tree such as a distinct icon or and a distinct name. These attributes display the type of the element, but these types are most often geometrical types (pad, cylinder, etc.) and not functional types (centering pin, draw bead, etc.). However the distinct representation of an object in the specification tree is an important idea to make it possible for an element to be identified. The identification is a good help for users but is also necessary if other people or processes have to find information (automatically or not) in a model. In addition, making elements identifiable is important in a CAD model but also in every other product representation.

Process and knowledge information
Process information can be integrated in a functional element so that it not only has meaningful characteristics in the current stage but also contains information for the adjacent stages.

Knowledge describes the internal characteristics of an element and its behavior in responding to external solicitations. In the following sections, we will specify the kinds of behavior there may be and particularly how we can implement behavior into a virtual product development environment.

4.3.3 External influences

4.3.3.1 Introduction

If we focus our attention on a given functional element and analyze all the aspects that can influence it, it appears that following factors have to be considered:

- User
- CAx system
- Standards
- Other elements
- Know-how
- Process chain

Each of the external elements has a certain interface to the functional element characterizing its influence. The following diagram illustrates these relations (Figure 4-5).

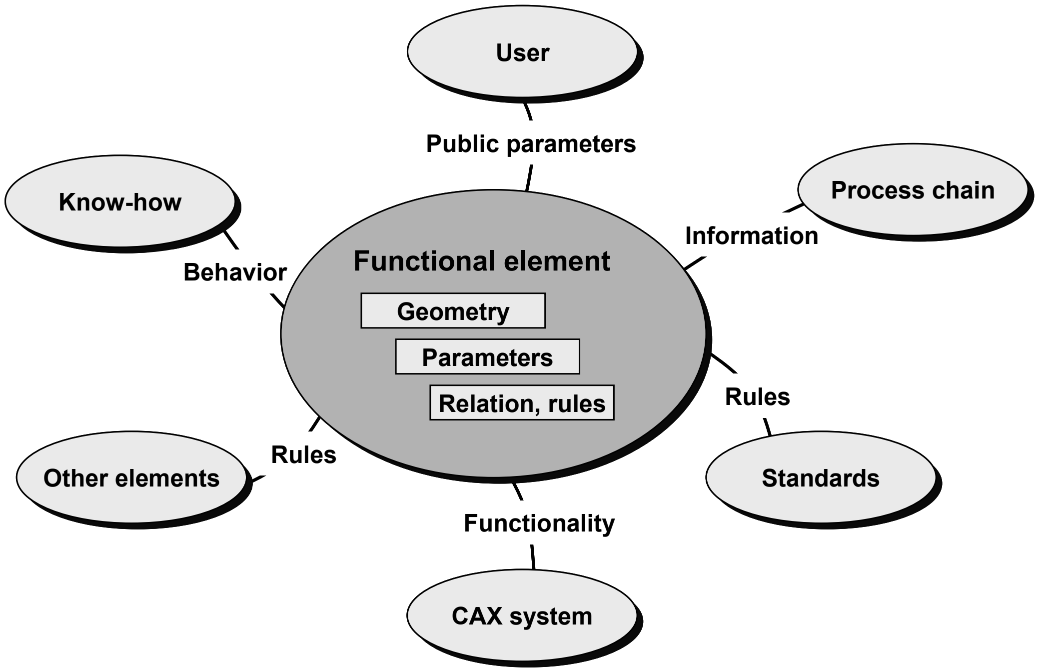


Figure 4-5: Functional element and its extern influences.

It is clear that the elements cited also have relations between each other, but the aim of the functional element is also to be an interface between all the elements that have an influence on the design. That is why there are cross-links between the know-how and the user or between the information about the process chain or the know-how.

In addition, the process chain is also connected to the other elements: not only standards but also the software affects the process chain. If the software had other capabilities than it has today, the process chain would not be the same as the data that have to be transferred from step to step could have completely different properties.

The difficulty here is to define what belongs to the functional element and what an external relation is: the internal behavior is strongly influenced by the external elements, making it necessary to define what should be included in the functional element (for example, standards, parameters) and what can be managed using external mean. In particular, when two functional elements have contact with each other, the interferences have to be managed to avoid conflicts. In conclusion, we have placed the functional element in the middle of our representation and we shall evaluate the connection with other elements, while also considering the net that connects all the elements.

Every object belonging to a component or its corresponding model can be represented as a functional element since everything that designers create has a certain function. However, all the given fields in the previous representation do not necessarily have to be completed. For example, a functional element can exist without being standardized or without having contact with other elements.

As mentioned, the environment of a functional element strongly affects its definition so that it is more effective to evaluate the environment before defining the functional element in detail. That why the following procedure is proposed to define a functional element:

- Identify what elements of the design have a functional meaning. The elements that can be considered as functional elements are generally items taken from the engineering language. No new element will be introduced because of the new modeling methodologies; the aim is solely to adapt the modeling methodology to the engineering way of thinking.
- Define all the factors that have an influence on the functional element. In this stage, the objective is to identify what has to be considered that can influence the behavior of the element.
- Specify the functional element. The specification of the element will be done in order to develop the behavior of the element in response to the last points.

In the following sections, we will come back to each point that has been touched on earlier to define its influence more in detail in the definition of a functional element.

4.3.3.2 User

Users are probably the most important item to consider in the definition of a functional element as they are the ones who will work with it: it is not sensible to define a technical solution which users cannot understand. We will discuss later the problem of user bias. What is important here is to note that, in each step of the definition of a technology, users have to be considered first.

Users are, in fact, the designers who will develop the product. They are the ones who control the functional element, that is, the people who give inputs to it. But they are also the ones who receive the feedback from this element, that is, the outputs. Even if designers give inputs to the element and manage it, there are some restrictions ensuring that they cannot do whatever they want. Here, we see the illustration of a connection with other outside influences: the restrictions imposed on designers by the functional element can derive from standards or the modeling system functionalities.

Other restrictions can simply arise from the fact that some modifications may result in the fact that the functional element does not have the same function anymore. *That is why it is important to develop technologies that give users enough freedom to innovate but at the same time clearly define what they are allowed to do*. Table 4-1 sets out the list of the actions that designers are generally allowed to execute on functional elements.

Action	Possible restriction
Insert	Not appropriate environment
Remove	Interdependences with other elements
Select the inputs	Input nature can be imposed
Define the position	According to standards and considering interferences with other elements
Set the parameter values	According to standards and design consistence

Table 4-1: User's actions on functional elements.

Users should at all times have the possibility to insert or remove a functional element. It is clear that the insertion is not possible in some contexts, particularly when the required inputs are not available. Removal of an element can also be delicate if other elements relate to it so that the relation is broken and an impossible situation occurs.

Selecting the inputs is a necessary operation to insert an element. The aim in this operation is to adapt the element to the context. The setting of the parameters is necessary to define the dimensions and the characteristics of the element. They are often standardized so that their value has a defined range or results from the combination of other parameters.

The selection of the inputs and the setting of the parameters are operations that can be done during the insertion but have also to be modifiable afterwards. In line with an order or a given situation, users can receive feedback from the system relating to the functional element. That is what is summarized in the following table:

Reaction	Explanation
Automatic reaction	The element adapts itself to the situation without asking the user
Information or warning	The user is informed of a situation but no reaction is expected
Ask for information or input	The user is asked to give an information to solve a situation

Table 4-2: System feedback.

The automatic reaction appears when there is no ambiguity in the situation. In this case, the system knows what to do and reacts without asking the user. This is only possible when exhaustive knowledge is related to the element and when this element reacts exactly as the user would have done it in the same situation. The reaction can be a modification of a parameter or of the position of an element. For example, coupled parameters have sets of values that have to be respected and when users change the value of one of the parameters, the other is adjusted automatically.

Information or warnings are issued to point out a problem or inform users that of a certain situation. This can be done when no action has to be taken or when the system does not know what to do. Thus users are informed and have to solve the problem by themselves. Information cannot only be issued in the form of text message but can also be represented directly in the geometry. The possibility of asking the user for information can be used when the system has detected a problematic situation and has a set of solutions that the user has to choose from. The answer can be in form of a parameter value or a modification of any attribute of the element.

Visualization

From the user's point of view, a functional element has to be visualized in the geometry field as well as in the structure tree so that it can easily be identified, selected, and manipulated. Model structure trees display the structure of the model with a hierarchy between the elements. Most of the time, each element type has its own symbol or icon and a standardized naming. This property has to be kept for functional elements in order to help users to better understand the model structure.

4.3.3.3 CAx Systems

The systems are important for the definition of a functional element. The definition of the functional element in a real context cannot be made without paying attention to the supporting systems.

The following points have to be considered:

- System functionality
- Systems performance
- Communication ways between systems

Some points have already been cited in the last section concerning the user requirements. Systems are the means to concretely implement the expectations of users. Thus the CAD system should provide the following options to the user:

- Insert a functional element
- Display the functional element in the geometry field
- Display the functional element explicitly in the structure tree
- Make the functional element editable
- Make the functional element suppressible

Other general demands have to be met by the systems:

- Enable reusable elements to be employed
- Archive the functional element
- Archive the models
- Provide a link between the applications

4.3.3.4 Standards

As they are engineering objects, most of the functional elements have corresponding standards. Standards define the shapes, dimensions, the position, or the number of the elements and are information that is permanently considered by designers. Either users know the standards as they use them frequently, or they consult the standards for some support: usually standards are archived in paper form or available in the intranet of the company but there is not direct link with the models that designers are working on.

Standards have to be respected and the CAD models are inspected at the end of the design to verify if it is the case. However designers not only check their model at the end but also consider the standards from the beginning and during the whole design phase. *Standards have to be considered particularly when a new element is added in the model and while proceeding to a modification.* That is why it is important, first, to make the standards available during the complete design stage and, second, to make them directly accessible in the part model or linked to it. This way of working would have two main advantages:

- Make the functional element reacting corresponding to the standards. In this way errors are avoided and users cannot create something that is not allowed by the standards.

- Inform the users of the existing standards without leaving the current application (that is, without looking at the intranet or standard book).

The first solution is to integrate the standards in the definition of the functional element so that it contains the information it needs to react to the users' intents or to the context. We have called the way the functional element reacts to the context in function of the standards its behavior.

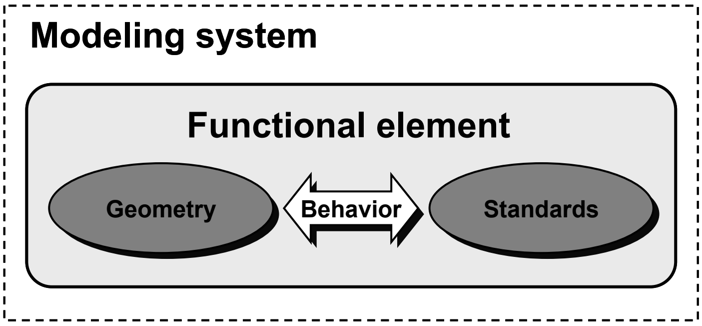


Figure 4-6: Integration of standards in the functional element.

The advantages of this method are that the functional element can work autonomously and does not need external references. The second solution would be to link the element to an external database that contains the standard information.

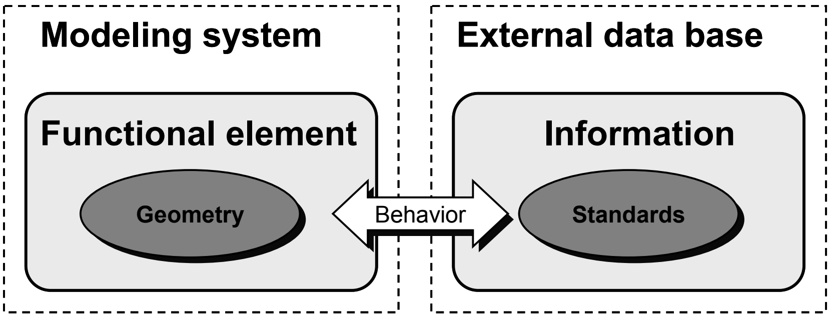


Figure 4-7: Use of external database to store the standards.

The advantage of this solution in comparison to the previous one is that the information relating to the standards is stored only once for all the elements of the same type and refreshing of the information can be done centrally. The difficulty is that it is necessary to insure that there is a link between the elements and the corresponding standards.

4.3.3.5 Other Elements

As a product model is a sum of several elements, it is necessary to manage the interactions between the elements and to define the behaviors when they are confronted with each other. The problem of collisions crops up when an element has an effect on another or when the relative position of two elements has to be defined. These situations are identified in more detail below:

- Contact with another element of the same part. If there is a contact between two elements in the same part, it is generally due to the fact that one of the elements is the input of the other one. For example, in the case of a panel with a hole, it is clear that the panel is the support of the hole. Thus it can be seen as the input of the hole.

- Contact with an element of another component. The contact between two elements of two different components has a functional meaning as it has an influence on the relative component positioning and then defines the behavior of the assembly. Thus the two elements define only one function. Consequently, both elements cannot be treated separately.
- Relative position with another element. The relative position is a constraint specifying that the elements have to respect a given distance between each other.
- Relation with another element. Relations concern not only the relative position of two elements but also the dimensions or other geometrical properties. In other words, the definition of one element can be defined depending on another element. This happens when two elements have the same function or belong to a group of elements with a global function.

The problem is to define whether an element is the master or the slave of another or if both elements have the same importance.

Master/slave relation
In this case, the slave obeys the master; that means it adapts itself to the master behavior. If the master is modified, the slave is modified respectively.

Elements with the same importance
When two elements have the same importance, it is not possible to define which one is the master and which one is the slave. In this case, the solution is to find out if both elements have a common third master element or if it is possible to define both elements as a unique greater functional element.

Managing the interactions between elements
The interactions between elements can be defined in view of standards or designer know-how. In both cases, it is necessary to formalize these interactions and to implement them. According to what is stated in the section on standards, it is possible to integrate the behavior in the elements themselves or in the external support.

4.3.3.6 Know-how

Definition 4-4: Know-how
Know-how is the knowledge of how to accomplish a task.

Know-how is a large set of rules that designers learn during their on-the-job experience to apply in their work. Know-how is different from other kinds of knowledge in that it can be directly applied to a task. In the context described in section 4.3.3.1, know-how defines how the elements have to react and interfere with each other, thus regrouping ideas that have already approached in the last sections. Actually, the interferences between elements follow rules that correspond to the designer's know-how and the standards are also a kind of know-how. Further sections will be dedicated to the implementation of know-how in the modeling process.

4.3.3.7 Process Chain

The relation of a functional element to the process chain is an important topic as it is a domain with a great benefit. The main advantage of a functional element is the reuse of the existing elements. This is valid for the geometry modeling as well as for the generation of other information models. Information models are deployed during the complete process chain, and the CAD system is only one link. There is information that leads to the CAD model and information that is derived from it.

Thus it is important that functional elements have a transversal definition across all the steps of the process chain or that they can be interpreted by each step. In this way, different representations of functional elements have to be developed and coupled to enable a continuous process chain and necessary information relating to following process steps have to be inserted in the current one.

4.3.4 Defining Functional Elements

Functional elements insure designers will not begin their design from scratch but will use these objects. Thus functional elements have to be developed before being used in a new project. This task could be done by any designer; but, to improve the efficiency of functional elements, they have to be standardized and thus they must be the same for all the designers. As a consequence, the development of functional elements is done centrally and they are classified by type and saved in a library that each designer can access.

The first step of the functional element development stage is the definition of its function. Afterward, the corresponding shapes are defined as well as the modeling sequence necessary to obtain them. In this step, the element inputs also have to be identified. They are the geometrical elements that are necessary to insert the functional element. For example, to insert a hole as functional element in a model, a center-point and a reference surface (and eventually a direction) are needed. Users will be asked to select the inputs during the insertion.

The functional element complexity may be variable. As a consequence, the representation of functional elements in the structure tree is important. In the case of complex trees with several components, they are represented as an assembly, in the same way as designers would have done so. However there is also the possibility to create closed elements to make them unchangeable, if they are standardized.

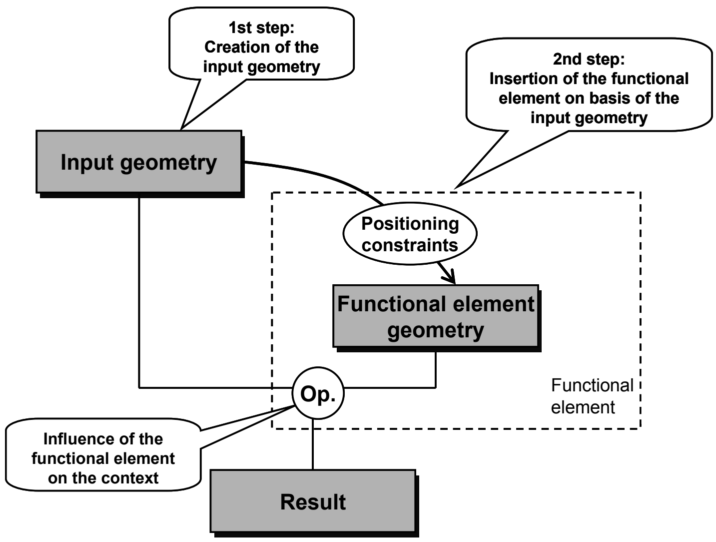


Figure 4-8: Insertion of a functional element in a model.

From the point of view of the geometry, a functional element yields the same result as elements that are modeled manually. That means that it influences the context by adding geometry to the existing geometry (Figure 4-8).

4.4 Knowledge in Functional Elements

4.4.1 Aim

Definition 4-5: Knowledge
Knowledge is the awareness and understanding of facts, truths, or information gained in the form of experience or learning. Knowledge is an appreciation of the possession of interconnected details, which, in isolation, are of lesser value.

We have mentioned all the conditions having an effect on a functional element and defining its behavior. Most of the time, the behavior of an element is the formalization of a part of the designers' knowledge.

Definition 4-6: Behavior
Behavior is the action or reaction of an element in relation to the environment or surrounding world of stimuli.

The behavior of an element defines how it should react regarding a given situation or event. In a traditional modeling methodology, the geometry is not active. This means that only designers define how to adapt a model to make it consistent. They know what the relations between the elements and between one element and its environment are and how to arrange them during the first modeling phase but also how they have to be managed in the case of design modifications. Thus all these rules are formulated in the mind of designers but are not implemented in the model itself.

CAD systems offer users the possibility to integrate knowledge into their model. In this way, the models become active, allowing greater versatility for modification. Figure 4-9 illustrates the comparison between the traditional modeling methodology and a modeling methodology based on the use of knowledge. In the first case, modifications could take a long time since they could lead to the changing or the remodeling of many elements. In the second case, the first modeling step has to be completed by the insertion of knowledge so that the advantages are not immediately visible but appear only later when modifications have to be made.

Modifications are a task that can take a long time. However the insertion of knowledge is also time consuming so that designers, despite its advantages, are unwilling to do it. That is why we will evaluate not only the possibility to insert knowledge into a CAD model but also the possibility to reuse it in connection with functional elements.

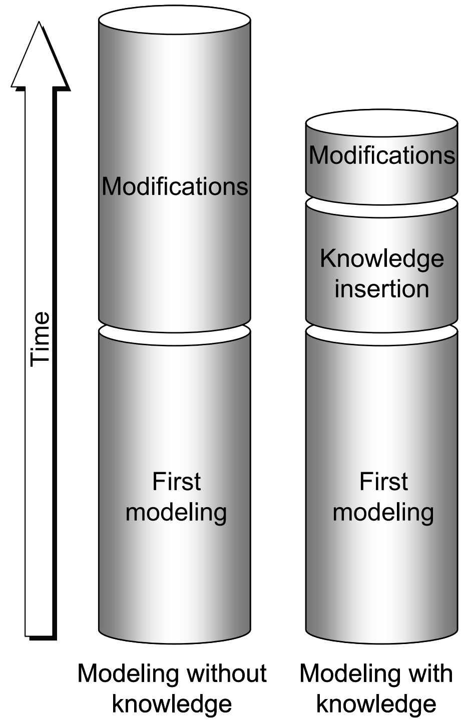


Figure 4-9: Schematic representation of the time gain by using knowledge in a model.

Quality and efficiency in the design process

The insertion of knowledge in the models not only helps to make modifications quicker but also aids in avoiding errors and suppressing iteration loops. In fact, the behavior of the functional element insures that the engineering rules and standards are respected during the manipulation instead of merely checking the model and correcting it.

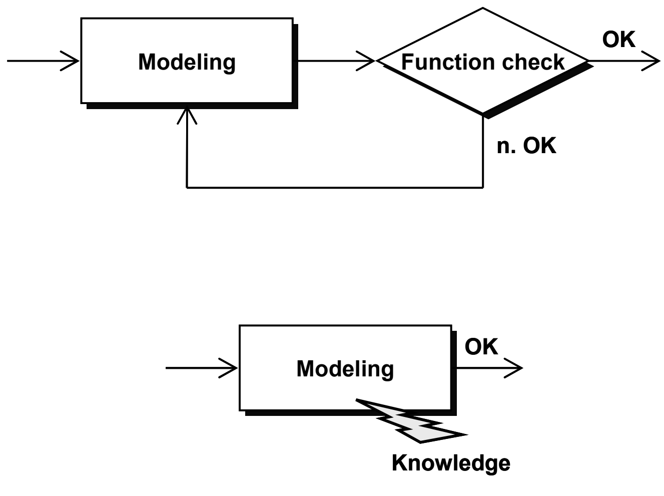


Figure 4-10: Model correction vs. knowledge integration.

In Figure 4-10, the two types of working are represented. The workflow at the top illustrates the correcting loops imposed by the delayed check of the modeling validity. In the lower picture, knowledge makes it possible to continually check whether the geometry is correct and particularly if the function of an element is respected.

4.4.2 Formalization of Knowledge

The aim of this section is to define how the set of rules that designers apply in their daily work can be formalized and integrated into functional elements. Most of the methods presented in the following are already known, but the goal is to identify the ones that are useful for functional elements.

Origin of the rules
The rules that a product has to observe take their origin in many different sources, some of which are set out below:

- Standards
- Mechanical working of the product
- Mechanical properties
- Manufacturability
- Cost
- Styling

We can note here that the sources of the rules are not always independent. Most of the rules dictated by the standards come out from the others categories. For example the standards often refer to mechanical properties or manufacturability of the product. Another example is the dependency between the costs and the manufacturability.

The rules that define the mechanical working of the product, independently of whether or not they are standardized, aim to check the presence of the necessary elements of the product, their form, their dimensions, their number, their arrangement or related position. The rules relating to the mechanical properties concern rather the dimensions and the shapes of the elements. This type of rule refers to the rigidity of the product or its ability to be found when needed. The costs are a function of the properties and dimensions of the product as well as of the quality. They are largely dependent on the tolerances linked to the standards.

The requirements derived from the styling particularly affect the shapes and the dimensions of the product. The styling does not concern all types of product and, as it is really a matter of taste, it is hard to formalize in concrete rules. Thus we will remember that rules affect a variety of factors:

- The presence and the number of the elements
- The shapes of the elements
- The position and the orientation of the elements
- The dimensions of the elements

While designing a product, there are elements that must be inserted in the product and in a given number. This can be a number of screws, holes, stiffener or anything else to be found in a given product.

As these kinds of rules refer first of all to the product, it is not necessary to include them in the element but rather in a higher level. This means that a rule has to be inserted in the product model itself or has to be checked after the modeling stage. That is what is done in the conventional design process where the validity of the model is checked later. The model can be validated automatically using macros whose aim would be to look for elements in the model and determine whether or not the number is correct.

If we wish to insert the rule in the product model, a parameter has to be used to check the item number of the elements. This parameter is either an existing parameter or an added parameter. As illustration we can consider a slab that has to be fixed using screws. The number of screws and corresponding hole depends on the length of the slab.

The three possibilities to control the number of screws and holes are as follows:

- Link the hole number parameter with the slab length, which would be a kind of "master parameter"

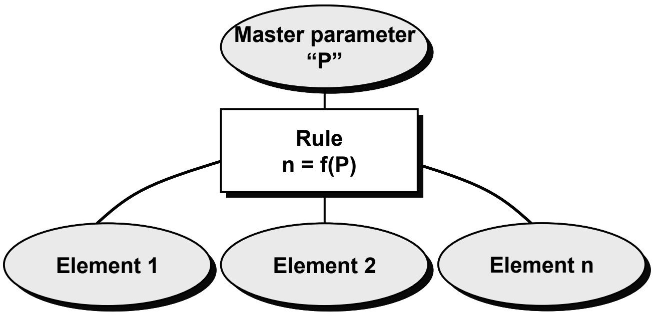


Figure 4-11: Master parameter.

- Create a free parameter that checks the hole number directly

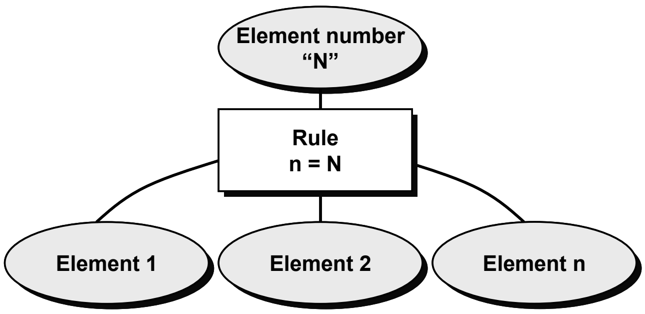


Figure 4-12: User parameter.

- Check the design afterwards

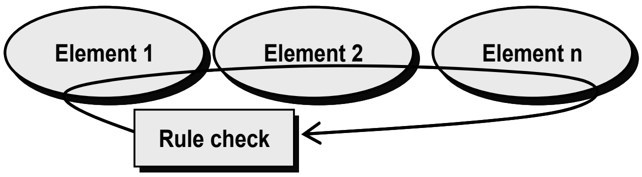


Figure 4-13: Rule check.

Depending upon the rule priority, one or the other of the possibilities will be used. The first option has the advantage of being full automated but does not give any freedom to users, whereas the second alternative gives them the possibility to choose the value of the parameter. The third variation can be combined with the second but has the disadvantage that it does not avoid errors but rather corrects them.

Shape of the element
As we wish to use functional elements as a reusable element, they will be available as objects whose shape is already established. Thus the standards are observed by the fact that the shape is given and that only allowed parameters are public. In the case where there are several possibilities of shapes for one element, all the variants have to be included in the element and driven by a parameter. This is possible as long as all the form variants do not affect the function of the element. Otherwise a new element type has to be created. In contrast to the previous section, the behavior can be implemented inside the element as shown in Figure 4-14:

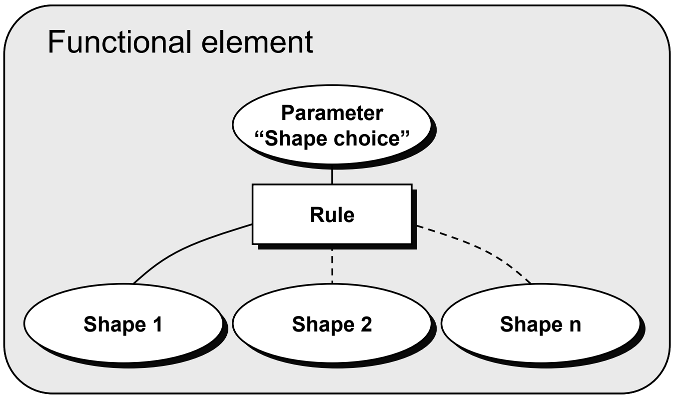


Figure 4-14: Selection of the element shape using a parameter.

Position and orientation of the elements
The position and the orientation of an element can be determined with the model coordinate system as reference or with other elements as reference. Moreover it is possible to use coordinates or constraints.

The choice of the methodology depends on the element function and, in the end, on the way it is defined in the standards. In other words, the definition of its expected behavior leads to the choice of the method for its positioning. If the position of the element is defined by the standards as a coordinate set, the solution is to parameterize the element relating to the part axis system or to position the element on a reference element (Figure 4-15). The reference element is itself positioned with coordinates referring to the axis system. This solution has the advantage that the position of the reference element can be previously defined so that rules can check it before the functional element is inserted and the interference between the functional elements can be managed by managing the reference elements. For more details about reference elements, see section 5.4.

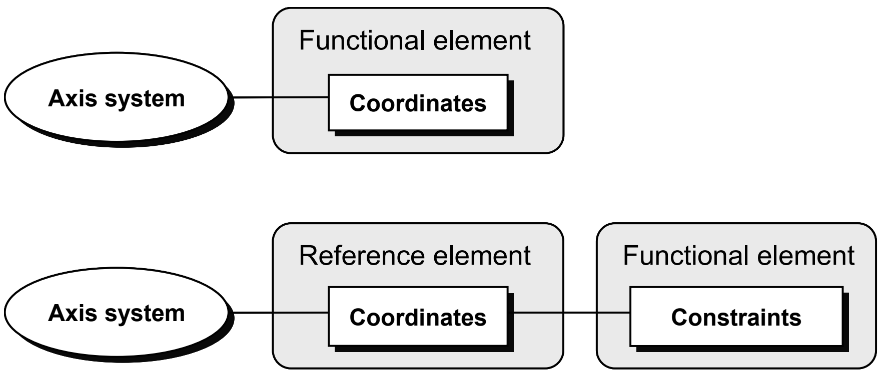


Figure 4-15: Positioning of a functional element.

However an element often relates to other ones. This means that its position depends on other elements and is defined by means of measurement, angle, coincidence or other factors. In this case, the position is a relative position but it works in the same way as with a reference element except for the fact that the reference can be any other functional element. A common utilization of such a method can be found in the definition of the distance between several items of the same type (holes, ribs, etc.) or when an object is positioned not relating to the center of the part but relating to an edge or a face of another element.

Both methodologies of positioning can be combined for the same element. For example, in one direction the position is defined by means of a coordinate and in the other two with constraints by means of a reference element.

Dimensions of the elements
In some cases, the positioning of an element can have influence on its dimensions. For example, when two faces of the elements are positioned relating to a third object, their relative spacing can vary, making the dimension of the element vary as well. But, most of the time, the dimensions are internal properties of an element. What is important here is to define what the main dimensions are in order to reduce the number of parameter values that users have to set. In fact, when several parameters are linked, it makes sense to publish only one parameter and to insert automatisms that adapt the other ones.

As example, we can consider the modeling of a hexagonal nut. To model a nut, several dimensions have to be given to the system, but in the mind of the designer only one is important: the diameter. Consequently the diameter of the nut has to be the main parameter that users have to set and the other dimensions have to be adjusted automatically. This adjustment is an internal behavior of the element that is hidden. Figure 4-16 portrays the three types of parameter:

- Public parameter: the diameter which can be set by the user
- Constant parameter: the angle between two edges of the hexagon
- Adjusted parameter: the length of an edge of the hexagon which is function of the diameter

The second diagram of the illustration shows the internal relations in the object. From the designer's perspective, only the public parameter and the geometry are visible. It is possible to impose restrictions on the parameter value so that only values allowed by the standard can be entered. There is also the possibility to display locked parameters as information. This lets users see the value of the parameter but they cannot change it.

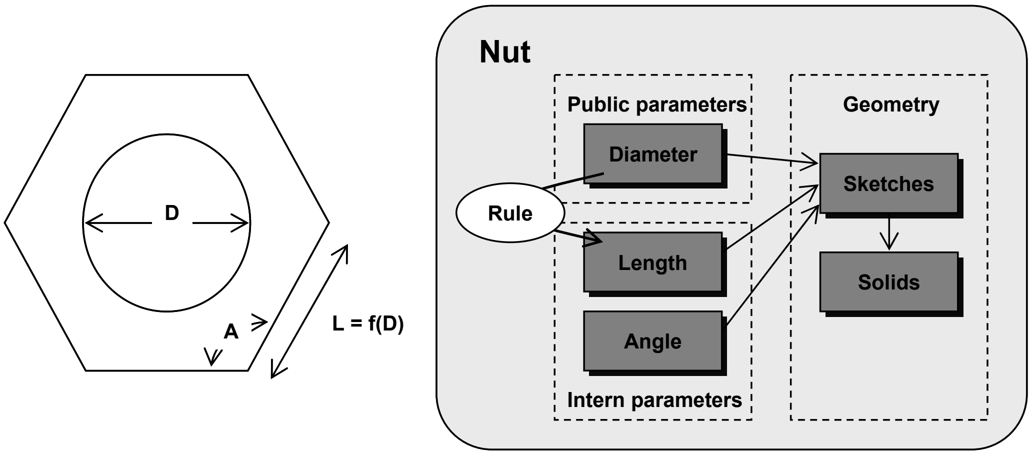


Figure 4-16: Example of a nut.

Summary

The previous sections have underlined the requirements of the integration of knowledge in functional elements. To summarize we will give a global description of an element structure. From the user's point of view a functional element has:

- A function
- A result geometry
- Inputs
- Public parameters

The specification of the functional element is another task that should not be done by the same person (more details in the section 8.4.2.2). This task is to develop the functional element so that it works as needed. The following additional points have to be considered:

- The result geometry as well as the intermediary steps
- The knowledge standing behind the geometry
- The internal parameters

The connection between all the elements to consider is depicted in Figure 4-17 with the specification of what users can access and what is hidden (internal working).

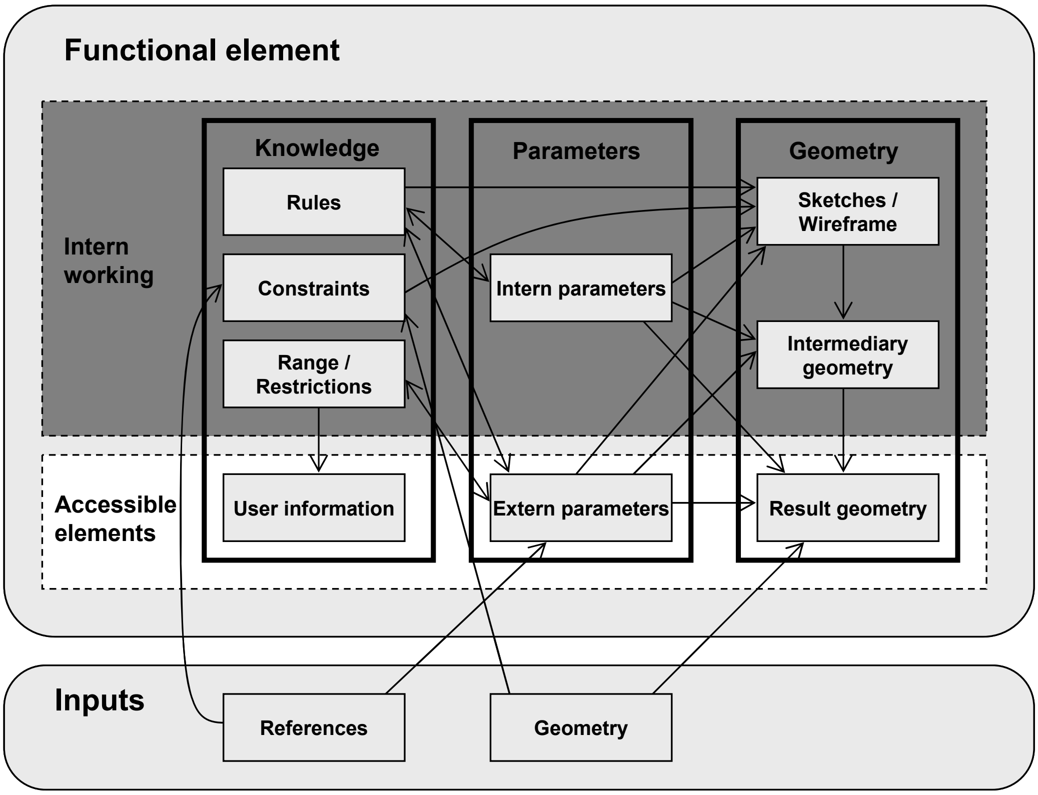


Figure 4-17: Structure of a functional element.

4.4.3 Implementation of the Knowledge

Parameters

As mentioned, the knowledge related to an object is a set of rules that describe its behavior. From the designers' point of view, these rules are explicit or tacit information that can be formalized largely by sentences. The problematic here is to translate these sentences into a language that can be understood by the CAD system. CAD modeling tools integrate knowledge language with the possibility to create rules and formulas. Most of the time, these functionalities refer to values of parameters. This means that parameters will be the starting points of knowledge and most of the rules will refer to it. This way of working makes sense, as parameters are the basic controlling elements of a model.

Parameters refer not only to lengths of segments or basic dimensions of the geometry but also to the activity of features, to their weight, their number, etc. In fact, parameters can be of any type: length, real, mass, volume, density, or any parameter belonging to a mechanical product as well as parameters of type string or Boolean. Parameters are automatically generated by the system when a feature is created. For example, a feature of type pad created by an extruded rectangle profile has the following parameters (Table 4-3):

Parameter	Type	Value range
Activity	Boolean	True/False
Direction	Boolean	Normal/Inverted
Width	Measure	>0mm
Length	Measure	>0mm
Depth	Measure	>0mm

Table 4-3: Parameter set of an extruded pad.

Formulas
The first kind of support for knowledge implementation is the use of formulas. This is a basic functionality that allows a given relation between several parameters to be fulfilled at any time. For example, given three parameters P1, P2, and P3 of the same type, a formula insures that a mathematical relation between the three parameter values is maintained: P1=f(P2,P3).

The application of such formulas is widespread. They are used when a constant relation between several parameters has to be assured. By changing the value of one of the variable parameters, the values of the parameters concerned by the formula are automatically updated as well as the linked geometry.

Rules
As do formulas, the rules influence the value of parameters by valuating a formula. The difference is that this formula is valuated only under given conditions.

The rules are of the type "if condition then action", where the condition refers to a given parameter of the geometrical model and the action sets the value of parameters. As a consequence, the value of the parameters to be set does not respond to a constant formula but depends on the context. Another aspect of interest proposed by some CAD systems is that users may decide when the rule has to be checked: by update, after a modification, or continuously.

Checks
The checks control the validity of relation. It works in the same way as the rules except that the reaction is only a message to the user. Thus when a check is not respected a warning is displayed and the cause of the non-respected relation is output.

Macros
Macros are a means to initiate actions that cannot be expressed using a simple relation such as formulas or rules. Indeed the limitation of both methods cited is that they can only set parameter values. If a geometry is to be created, deleted, or greatly modified, macros are useful.

Design tables
Using design tables, it is possible to the store stets of values relating to parameters. This is useful to link standards to a given design. The use of a design table makes sense when it does not belong to the model but rather when the model refers to the design table stored in a central place. In this way, several models can access to the

same design table, and when a modification of the standards has to be implemented, a modification of the central design table is sufficient.

4.5 Conclusion

This chapter has presented the concept of functional elements that are a means to support any product development process. They are inspired by the idea of design features but propose a larger field of application. Thus we can consider that features are a sub-category of functional elements. Functional elements are first intended to support product designers who need methods suiting their way of thinking. To fulfill this requirement, functional elements are adapted to the domains they are employed in and given properties which could vary from company to company, depending on their specificities (for example, standards, manufacturing equipment, knowledge, processes).

In the past years, there have been elements which had the property to be reusable. Thus these caused the design and modeling processes to change, as it was necessary to develop these elements before using them. The task of developing the elements to use them later in the design phase has a particular importance since it determines the success of working with function elements. It is hence an investment whose benefit will increase with the frequency the elements are used.

In addition, functional elements make the link between the different steps of the considered process chain by being a central information carrier. They contain behaviors making them adaptable to the context, supporting designers in their task and displaying necessary information about adjacent processes.

Chapter 5
Application of Functional Elements

5.1 Functional Features

5.1.1 Definition

There are many definitions and surveys about features (refer to section 2.2). In this thesis, however, we will consider features as a sub-category of the functional elements and will employ them on basis of the definition set out in [MbSH-02b]:

Definition 5-1: Intelligent features
Intelligent features are consistent and semantic objects that reflect both passive knowledge (experience and usage) and active knowledge with respect to the geometrical operations (dynamic behavior during interaction with other features) and the engineering reasoning (executive procedure for design consistency and for automatic mapping to related phases of the product lifecycle).

This definition is adopted in this thesis. Yet we will also consider, as mentioned in [OvJa-94], that features are constituents of a part unable to exist on their own and that they fulfill a basic technological function of the product.

Definition 5-2: Functional features
Functional features are intelligent features that represent one indivisible mechanical function of the product.

5.1.2 Feature in the Modeling Stage

As a product takes form first in the design stage, this is the first aspect to consider for the definition of features although they may have a continuous meaning across the whole product lifecycle. Practice shows that product designers concentrate on their work and do not have or do not want to have a global view of the downstream process chain. Their objective is to create the product model and they are not willing to carry out additional tasks that do not concern them directly. It is hence difficult to require designers to respect the requirements of the manufacturing process, for example, and thus machining features do not interest a designer. Consequently, it is important to develop features that support designers themselves and that are improved for this task. The other steps of the process chain are important but they do not have to hinder designers. It is the task of the feature developers to make features that are optimized for each step without increasing the complexity of each of them. Thus this chapter first deals with the requirements of the modeling stage. The other aspects will be treated later.

A feature is the smallest element that designers should have to consider during the modeling. "Smallest" means that a division of the features into sub-element is not necessary and does not make sense for designers. Thus, a feature is an elementary element of the design stage. But, in our definition, a feature is an object more complex than the standard form features of CAD systems such as fillets or pads.

A feature can be as complex as needed as long as it fulfills only one function that cannot be divided into sub-functions. The analysis of concrete examples has yielded the following information on features:

- Features are repetitive objects. The same feature may be found several times in the same model and the same features may be found in various models
- Features are most of the time standardized. Objects that are deeply present in the engineering environment are very often defined by standards
- Features are specific to each domain of engineering. Each domain has its own language, knowledge and habits
- Features cannot exist on their own. Features need other elements as support; otherwise they would be confused with a part. An element could be both a feature and a part at the same time only if the geometry of the part is very simple (for example, a pin or a disc).
- Features have few master parameters. Although a feature can have many dimensions, it is preferable and possible to define few parameters controlling the whole geometry of the feature.
- Features can be a link or a contact between several parts. There is a whole category of features whose function is to assemble, guide, or position parts.

From these notes, we can deduce that the shapes of the features are constant. However there are smooth variations depending on the context and the value of the parameters. In other words, a feature is an object whose properties are strongly defined and thus users will have restricted possibilities while using features. Hence, a feature will generally be a closed object where only the public parameters and the result geometry are visible. The intermediary steps of the modeling will be hidden.

Adjustment to the context
One of the interesting aspects of features is that they are objects that should support users in the modeling stage. To reach this goal, features should contain mechanisms that make them adapt themselves to the context. *While it is inserted, a feature analyzes the inputs given to it by the user and adjusts itself.* This adjustment is necessary to compensate the variations of the inputs from one design to other and to make the model consistent. Such mechanisms are already implemented in CAD systems, for example, the through holes, which automatically do the following:

- Find the direction of the material to make the hole being located in the input solid and not in the opposite direction
- Adapt its length to the thickness of the solid so that it goes through it

This adjustment works during the creation of the hole and at each model update as well. For more complex features, the adaptation to the context is deduced from the designer's experience as the designer knows how the feature has to react.

Consistency of the inputs
To enable features to fulfill their functions, the inputs have to be compatible with them and it is their role to check that these inputs are correct. For example, a feature of type blind-hole can check whether the input solid is thick enough to receive it. If the depth of the blind hole were greater than the solid thickness, it would not be a blind

hole anymore but a through hole. The reaction to start in this context depends on the behavior desired by the user: the depth of the hole could be automatically reduced to insure a minimal thickness at the bottom of the hole or the solid thickness could be increased or users could be informed.

Shape and solid features
As there are solid and shape models, there are also solid or shape features. The modeling of shapes has two possible applications. The first one is the modeling of components that are so thin that they can be considered as a surface without thickness, for example, car panels. The second application is the use of surface modeling as an intermediary step to the solid modeling when the shapes are too complex to be generated with prismatic geometry. The aim of a shape feature is to add a shape to an input surface or to trim a part of it. Solid features are composed of a set of volume addition and subtraction operations that modify the input solid.

User feature and specification tree
Today CAD systems are feature oriented. The features proposed by the systems are basic features and not features in the sense we have defined them in this thesis. Additionally, the systems do not consider the specificities of all the various industrial domains. Thus the use of features lies on the possibility for users to create their own features, called user-features.

Basically a part is the result of subtractions and additions of simple volume. Each operation between two volumes leads to an intermediary result that can be combined with other volumes and lead, step by step, to the end result. An example for the representation of a solid with this view is set out below:

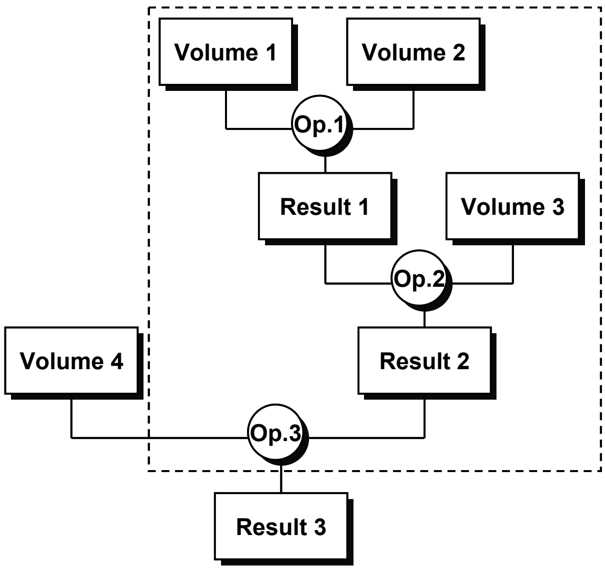


Figure 5-1: Logical tree.

We will call this type of representation a logical tree. However, the representation of the same structure in the CAD systems takes another form. The diagram below shows the same part as previously but with the usual CAD structure tree view:

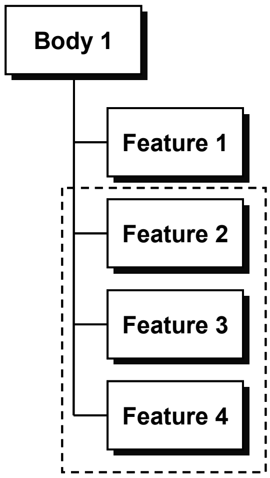


Figure 5-2: Feature oriented representation.

In this example, the representation is that of a feature-oriented system. Each of the features is the combination of a volume and a logical operation, and the body placed at the top is the last result. As there is a chain of logical operations, we have to consider all the operations to determine if a volume is, in the end, a positive or negative volume. The following applies:

- Feature 1 corresponds to Volume1 and the Operation 3.
- Feature 2 corresponds to Volume 1 and the combination of Operations 1, 2, and 3.
- Feature 3 corresponds to Volume 2 and the combination of Operations 1, 2, and 3.
- Feature 4 corresponds to Volume 3 and the combination of Operations 2 and 3.
- Body 1 corresponds to Result 3.

User features contain a sequence of several standard features. In our example, if we consider that the sum of the features 2, 3, and 4 (or the corresponding elements in the frame of the first picture) is a user feature, the related structure tree is shown in Figure 5-3.

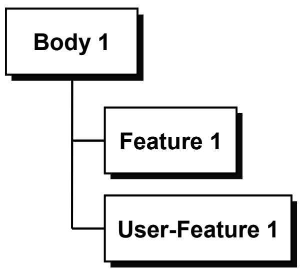


Figure 5-3: User-feature tree representation.

The question is now to first define whether it is possible to describe a whole part with features and second whether it is the ideal methodology. CAD systems are made so that it is possible to create any geometry with standard features. Everything that is possible using standard features is possible with user features since user features are based on standard features as well. As a consequence, an entire product could be defined with user features.

Yet, it only makes sense to employ user features if they have more functionalities than standard features. Although it is possible to entirely model a component with functional features, we have to evidence whether or not it is useful to do so. There are two problems that would be a restriction to the use of user features for the overall part description:

A. Practice problem
To model a whole part with functional features, an exhaustive functional feature library must be provided. In practice, elements that would be defined as functional features are elements that are often used in a given domain of application. If each element were modeled as a functional feature, the risk would be that the feature library would be enormous. Also, there are features that have different functions but the same shapes so that they will have to be modeled as different features. Therefore it would be difficult for users to deal with a feature library that contains so many items. Further, many functional features modeled with our methodology would not be different enough from a standard feature so that designers would not see the advantage of the functional features in comparison with the standard features and would not use them.

B. Definition problem
The second problem concerns the definition of functional features. A functional feature only makes sense as long as it keeps its function. The risk is that the combination of functional features deteriorates the function of some of them. Concretely if a functional feature is inserted on another one, it could modify it so that its function is no longer fulfilled. However only a feature on which another feature is inserted can be altered: a feature that is inserted at the end of the chain is not concerned. Thus there are two groups of features:

- Features that serve as a basis for the insertion of other features
- Features that are inserted on other ones without having further children

By analyzing some models created with the feature methodology, it appears that features modified by the insertion of other features are often simple shape features with the function to serve as support for the others. These features typically define the general shape of the part. In contrast, the features that take other features as a foundation are employed to detail the product model and thus have specific functions. We can hence define two kinds of features: support features and detail features.

As support features have no predefined shapes and can be affected by other features, we will exclude them from the functional features. Moreover we have defined features as elementary objects having only one function and, in many case, the supports have more than a single identifiable function. We will see in further sections that there are other methodologies to model this kind of element.

5.1.3 Features in the Process Chain View

After the insertion in the product model, features serve for the overall process chain. Therefore they have to contain process information or to be linked with process information.

CAD features with process information

It is not realistic and desirable to insert all the process information in the CAD model as this will encumber it and disturb designers. However some information is traditionally found in the CAD model. For example, the definition of tolerances belongs to the design stage and should be integrated into features. In the automobile industry, two solutions are employed to represent tolerances. First, the tolerance representation function of the CAD systems is used, providing the possibility to visualize tolerances in the 3-dimensional representation as was done before in 2D drawings. Second, the surfaces to be machined with a given quality are represented in color according to certain conventions. This way of working can be maintained with the feature methodology. The problem is then to know whether the tolerances are always the same for a given feature or if they vary. If they do not vary, they can be integrated in the feature.

But if a feature has various tolerance possibilities (depending on its function, for example) the tolerances have to be linked with a parameter that sets their value. In the case where there is no rule for the tolerance, the feature is created without tolerances, which are modeled later by designers.

CAD features linked to other processes by means of an external system

Despite the possibility to insert information directly in features, it is also possible to manage information using an application external to the system. Its role would be to create the link between the features in each step of the product lifecycle and to make only the relevant information available.

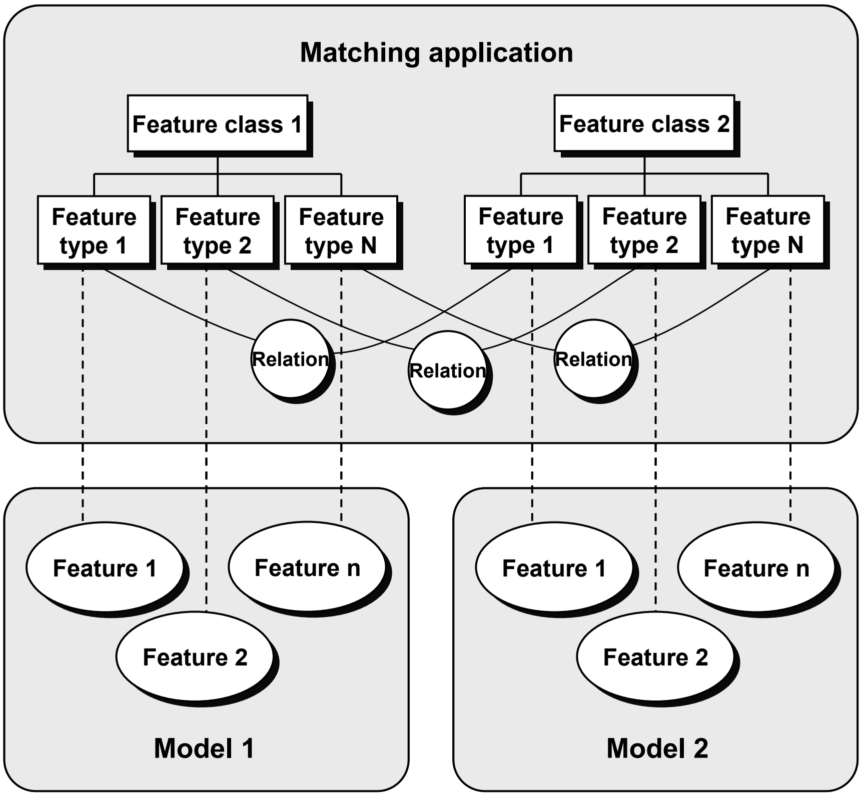


Figure 5-4: Linking of features by means of an external application.

Figure 5-4 shows the structure of a solution based on a matching application external to the system. In the first model, there are features of various types that have to be connected to corresponding features in the second model. The connections are made according to the feature's attributes and according to given relations. These relations among the features are not directly created from feature to feature but rather from feature type to feature type by storing information in the matching application [Zimm-05]. This solution has the advantage that it reduces the amount of information in each model and allows a central and unique access to the definition of the feature's links.

Multiple-view features
The previous section has treated the case where features of various model have to be linked by means of a given relation. In some cases, however, the models concerned are of the same type with one of them generated directly from the other. This occurs, for example, in the case of components that are to be produced by a forging tool. In the design process of such a product, tool designers employ the model created by component designers as input for their task. Hence, we could find features that are present in both models but with different shapes so that multiple-views features are needed. This kind of feature can be switched from one view to one other by selecting a parameter value.

5.2 Engineering Templates

5.2.1 Requirements

We have seen that a product could be described with features; that is, a product representation could be a sum of features. However this does not mean that these have to be inserted one by one. If we consider a product as a sum of features, alternative solutions have to be found to support the modeling of the product by using features not as isolated elements but rather as a group. In this chapter, the motivations to develop other methodologies than those based on features are given.

Efficient modeling
Features have already taken a step in the direction of efficient modeling. They allow objects to be inserted quickly instead of necessitating that they be modeled from the beginning. *But the insertion of features is still a repetitive operation as the same feature has to be frequently used.* Second, we have demonstrated that feature methodology is not geared for the creation of support elements.

As mentioned above, support features describe the main shapes of a product and it is difficult to define a common geometry that would fit all the possible cases and is controlled by only a few parameters. However, it is possible for several variants of a given product to identify similarities and common parts having the same function. Thus there are two requirements:

- Define reusable objects having common properties but with a limited standardization degree. In this case, it is important that designers have sufficient freedom to specify the object as desired. Thus the challenge is to find the right compromise between standardization and degrees of freedom.
- Define objects supporting the modeling stage and avoiding repetitive modeling steps. Designers would have the possibility to employ objects present in the library, leading to time saving.

Standardization
Standardization is the process of establishing and formalizing common rules and properties for a kind of items in order to improve the efficiency in handling them and regulating their interactions and cases.

In the product design process, there are often several people working on the same model or needing the same information. In the case of a complex product geometry or products made up of several parts, two designers frequently have to work on the same model. In such a situation, each designer is assigned one part of the product or one body of the part. The second case occurs more frequently. It happens when someone has to extract information from a model made by someone else. That is the principle of a process chain where the information created in one stage is used in the following one. Thus there are two types of collaboration: parallel collaboration and sequential collaboration. They call for availability of working methodologies and also for standardized product descriptions. This section focuses on standardization; the issues of collaborations will be handled later.

As soon as various people intervene in the same design project, standardization is necessary for many reasons:

- Quick access to information: The problem appearing nowadays is that each designer has an own way of working, an own modeling methodology, and therefore an own result presentation. Concretely both the design and the structure of the model vary from one designer to one other and it is difficult for one person to find the information needed when s/hee is not the one who has completed the model. Standardization of design and modeling methods would help to make the model independent of the designer who created it.
- Reusability: Standardization favors the orientation in the structure of the model, making later modifications easier and reuse of a previous model for a new application more comfortable.
- Quality: Standardizing the model representation and the methods makes it possible to reduce the number of errors and enhances model quality.
- Automation: Standardization of one element makes it identifiable by other applications. Thus it is possible to automatically extract information from a model to generate other information models (for example, to generate a manufacturing plan on the basis of a CAD model).

Information carrier
In product development, there is a need for an information flow between all the steps of the process chain. The information to be considered in the definition of the engineering template depends on the context and on its utilization, but generally it is information about the previous or following process stages.

5.2.2 Definition

The requirements mentioned in the section above clearly indicate that a new modeling methodology has to be developed. The solution proposed here is the use of engineering templates, whose definition is given as follows.

Definition 5-3: Engineering template
An engineering template is a reusable, parameterized, and structured element with continuous process information.

The definition mentions the term "element". A more precise word has not been used since, at this stage, we have not determined at which level of a product representation an engineering template is defined. We already know that an engineering template is bigger than a feature, but it could be a part of a product or the whole product.

For example, in the design of an engine, the engineering template could be any one of the following:

- The whole assembly (the motor) with all the components and details
- One of the components (the cylinder head)
- A part of one of the components (a cylinder)

It is clear that the ideal solution would be the one where the whole engine would be created with only one template that would quickly generate the entire geometry of the product. But is it possible to develop such a complex template? There is no simple answer to this question, but there is a general method based on the analysis of the product.

The aim of this analysis is to identify the reusable parts of the product. In our context, a "reusable" element is an element that can be extracted from a model and inserted into another one. The inserted element does not have to be exactly the same in both models but it must have similarities. It is therefore hard to define a template by analyzing only one product. To define a template successfully, it is necessary to have several models of the same product family.

In the automobile industry, despite the constant effort invested for innovation, there is a huge number of parts which can be found in a similar form in different models. Indeed there are various segments defining the car type, and each car of a segment is periodically replaced by a new one. Thus if we wish to create a template for a part of a car, for example, a side panel, we can compare the side panels of all the segments that are currently produced, or we could compare all the previous cars of the same segment (see Figure 5-5).

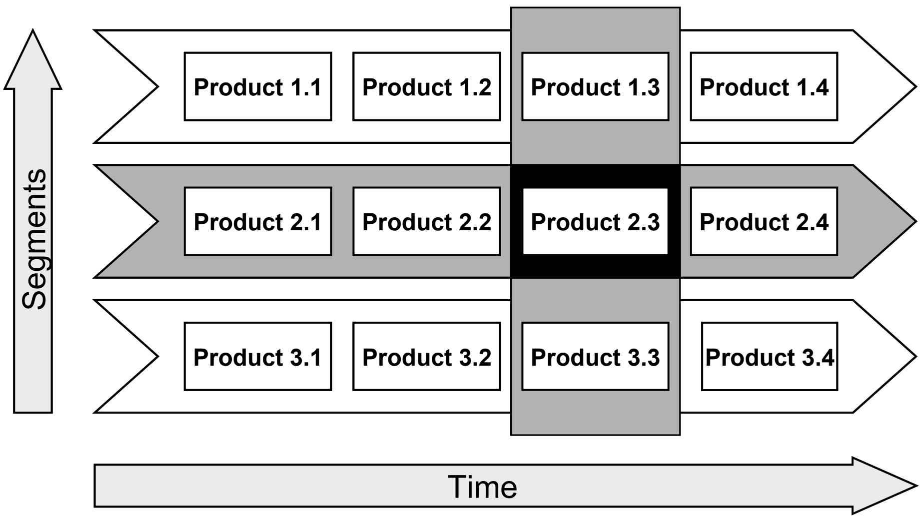


Figure 5-5: A product in its product family.

The analysis and the comparison of various models of the same product family can yield three kinds of results:

- Case 1: There is no similarity between the models: In this case, each product is a fully new one and it is not possible to reuse any part of the existing geometry. This situation is not favorable to the use of templates that are reusable objects. Nevertheless only infrequently does a product have a singular design that does not resemble any other.
- Case 2: There are several partial similarities between the products: several products can be globally different but composed of the same basic elements. In this case, not the whole product will be defined as templates but only the basic elements. Then several elements are necessary to create a product. As illustrated in Figure 5-6, various elements are common to two or three of the products. They can hence be modeled as templates and serve as modules for the design of the three products.

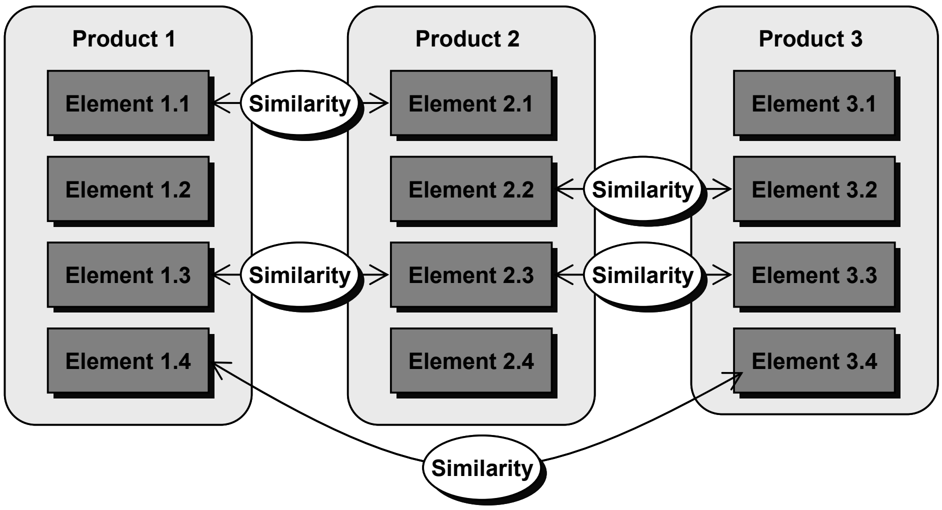


Figure 5-6: Similarities between the elements.

- Case 3: All the products are similar: when the products of the same family are completely similar, a single template can be used to describe all of them.

To illustrate this, we take a look at an engine. If all the engines were different, engineering templates could not be employed (case 1). But there are similarities between all the engine variants although they can have various configurations. For example, there are four-cylinder engines and six-cylinder engines in a "V" configuration. The arrangement and the number of cylinders are different, thus there are different solutions. First, we could create a template describing the whole engine with complex behaviors allowing it to cover all the possibilities. Although technically possible, this solution would be too complex to deal with and there would be too many parameters to set, so that we will exclude such a solution.

The second solution would be to consider both engines as belonging to different product families: the family of the four cylinders engines and the family of the V6 engines. By doing so, the problem now corresponds to case 3, where the whole product of a family can be modeled using one template. We have only reduced the room of the family. This solution implies more work as there would be more templates to create (because of the enhanced family number).

The third solution is to consider that both engines are not similar but that they have strong part similarities: for example, the cylinders are common parts present in both engines. Thus a cylinder can be modeled as a template and each engine would be composed of several cylinder templates. The disadvantage is that a product now consists of several templates and the relations and interactions between them should be managed.

In this example, we will keep the two last solutions: reduce the family room or split the product into several templates. The brief description given here is not sufficient to decide what the ideal solution is. To achieve such a solution, it would be necessary to carry out further investigations and to test the solutions in concrete applications.

Detailing degree

Once the size of the engineering template is defined (whole product or a part of the product), the analysis of the products of a same family should give further information such as the template detailing degree. In fact, the creation of an engineering template makes it necessary to find the similarities in a product family, but the similarity degree could vary. This means that products can have coarse similarities or finer similarities.

In the case of coarse similarities, the products are similar from the greater perspective, but the details differ strongly. A finer similarity means that the products are similar in details. The detailing of the template is partially done using functional features, that is, templates are feature-based object and a part of the model detailing is already done using the template (Figure 5-7).

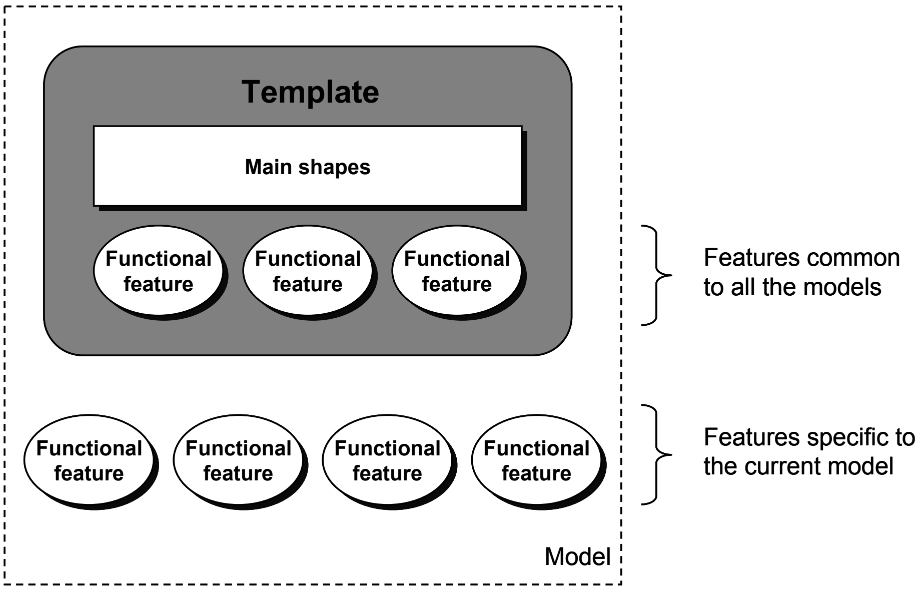


Figure 5-7: Detailing degree of engineering template.

Depending on the degree of similarity, the finegrainedness of the definition of the template is established. Concretely the template has to contain only geometry and information common to all the product variants. Everything specific to one product in particular has to be omitted.

Parameterization
In the definition of the engineering template the term "parametric" is employed. This section aims at defining the idea of parameterization applied to engineering templates.

Definition 5-4: Parameterization
The parameterization is the action of defining a set of factors whose values determine the characteristics or behavior of an object.

An engineering template is supposed to suit to all the variants of a product family. Despite the similarities between the products, variations are present and templates have to cover all of them. In other words, templates have to maintain a certain flexibility to match every context. The mean to make it possible the template to adapt itself to every context is the use of parameterization and knowledge as it was described in section 4.4.

Flexibility
Engineering templates are more complex than features and should have more flexibility. In the case of features, parameterization essentially aims at facilitating the work of designers, and, second, at insuring that the standards and the function of the features are respected. Thus the parameterization of features represents a support for users but also a restriction of the utilization options.

In the case of templates, the parameterization should maintain the role of user help but less restriction function. It means that template have to be "open" objects where users can access to every element of the definition. Thus, after the insertion of a template, it resembles a normally modeled object. The difference to the user's modeling of the same geometry is that there is much more knowledge and information inside it.

Moreover, if a template were a closed object, it would be controllable only by means of public parameters and, to allow this, a template would need so many parameters that it would be too complex for users to understand and employ it. For example, if a profile has to be set by the user, there are two possibilities:

- Lock the access to the profile definition and create public parameters to control it.
- Give free access to the profile definition without additional parameters.

In this example, it is clear that the number of parameters may be too high if the profile is complex and that an unequivocal name for each parameter will be hard to fix. Thus it is easier to grant users access to the definition of the profile, which is the way the user is used to. However free access to the definition of the elements is not an absolute rule. If information within the definition of the engineering template is relevant for the process or if it standardized, it has to be protected by introducing usage restrictions*. In conclusion, the utilization of templates in practice shows that it is important to make templates easy to use and to leave sufficient freedom for users but, if necessary, some information may have to be locked.*

Innovation
The aim of engineering template is to make some elements reusable and not to create the same design every time. It is crucial that designers are allowed to express

their creativity: thus a key requirement of the engineering template methodology is to permit innovation.

What should guarantee innovation? The possibility to set the value of the template parameters and the fact that templates are open objects should insure necessary degrees of freedom and thus innovation as well.

How should innovation be guaranteed? As mentioned above, everything that is not a general property of the template has to be kept out of it. But the analysis of the similarities between the part does not suffice to guarantee innovation since innovation, by definition, implies design of a new product, that is, to change what has already been done. What we wish to demonstrate here is that innovation and reusability do not represent opposed aims although this may seem so. *First, there are many product families where innovation lies in the optimization of existing design rather than in a completely new design.* In this case, the general design remains the same and the differences between a new product and a previous version depend on details and modification of the dimensions.

Second, most of the innovations do not concern only one isolated product. They are often implemented in several products of the same family so that there is still the repetition factor, which justifies the use of template. Third, when a product has to be newly designed, not all its constituents are replaced or modified but rather only key parts. This means that there is often a part of the product that is considered as constant, and the innovation resides in the modification of the rest of the product.

Finally, the innovations in product design are not just the responsibility of the designers that model the product. They are discussed in plenum so that it is clear before the beginning of the modeling that an innovation has to be introduced. The actors should hence previously discuss whether the available templates can be used for the new context and, if the template is not adapted, if it would be better to model the template before modeling the whole product rather than vice versa.

Structure
The definition of engineering templates describes them as a "structured" element. The template structure is one of the ways toward standardization, which has been discussed in section 5.2.1. Consequently standardization aims to simplify the understanding of templates by users and to allow a continuous information flow.

Process information
There are various types of process information. Many surveys deal with the integration of machining process information. As mentioned, engineering templates do not replace features but complete them. Thus features will keep their role as an information link between the design and manufacturing stages. In the case of templates, there is a need for information integration within the design process itself. In fact, in most of the design processes, there are various CAD models that affect each other. In parallel, other model types are created and influence the product model. In particular, there is often a separation between CAD and simulation models. As a consequence, engineering templates should contain information relating to the design process itself, which up to now belonged to the user's know-how.

5.3 Integrated Components

5.3.1 Why a New Category of Elements?

Most products are made of an assembly of several components. Components of a product can be specific but there are also a lot of components that are standard and are regularly employed. This section focuses on the standardized components.

We have introduced two kinds of functional elements: features and engineering templates. The question is whether both methodologies can be applied for the modeling of standardized component. *The standardization of the concerned components makes that they are precisely defined and that the differences between two components of the same kind are very slight. In fact, these differences do not concern the element shapes but rather their dimensions.*

As described above, features are elementary and standardized elements. The modeling of standardized components by features would be possible as each component is a sum of features; yet, due to the previous note about standardization, it appears that the decomposition of a standard component into features is not necessary for the modeling phase. Thus it is better to consider standardized components as a whole entity such as templates. Nevertheless a standard component is not a component that underlies innovation. In contrast to templates, standardized components do not need to offer much freedom to the user. Standard components are typically components taken from a given catalog and users only have to choose which one suits the context. Therefore features and templates are not the best way to model standard components. In addition, there are more factors justifying the introduction of a new category. The modeling of standard components takes lately place in the modeling sequence so that they are inserted in an environment where elements have already been modeled. As a consequence, standard components are not elements that can exist on their own: they require support.

The last argument is probably the most important: standard elements have a given environment as discussed in the next section.

Standard components and their environment
Every component has an impact on the model in which it is inserted - first of all, since it has to have a mechanical link with one or several other components of the product. The problem here is not to define the nature of the link between the components (this can be a fixation link, a translation, or a rotation guiding, etc). What is important is that a standard component cannot fulfill its function without being associated to corresponding shapes in the support. For example, a screw has no function if not associated to a hole.

Further, a standard component is often associated to other surrounding components. For example, a nut has to be associated to a bolt to fix two parts so that they cannot be seen independently of each other. Thus standard components have their environment. It means that they have a context of insertion and they have auxiliary components as well. Both the context and the auxiliary components are dependent on the main component and are influenced by it. We will call environment the set of elements surrounding the standard component and directly influenced by it.

Current modeling methodology
At present, the standard components are taken from a component library so that designers do not have to model them at each time. These components may be parameterized but what is missing is the integration of the environment. This means that each component is inserted as a single part or as a group but without active knowledge and influence on the context.

Therefore designers have to manually insert the components they need. They have to position it relating to the support and to create the corresponding shapes in the support. For example, if they need a screw, they have to position it in the model, to define its orientation relating to the support component, and to model the corresponding hole into it. In the case of modification of the screw, designers have to modify the hole manually so that the consistency of the group [screw + hole] is kept. This is valid for a modification of the screw dimensions, for a modification of the screw position, or simply if the screw is deleted.

5.3.2 Definition and Properties of the Integrated Component

Definition 5-5: Integrated component
An integrated component is a standardized component inserted with its related environment and active behavior.

The type of components we have introduced should have the same properties of the already available components: they have to be reusable and parameterized. What is added here to the working of such components is the fact that they ought to be accompanied by their environments. We hence call them "integrated components".

From the geometrical point of view, an integrated component is composed of the following:

- The main component(s): The main components are the components that justify the function. Without these components, the function is not fulfilled. Most of the time there is only one main component; but if the function is fulfilled by two or more components that have exactly the same importance, they are also defined as main components.
- The auxiliary component(s): The auxiliary components are those that are present to accompany the main component. They fulfill a function of a secondary importance in comparison to the main components and they cannot exist if the main components do not exist.
- Additional shapes: As mentioned, integrated components have an influence on the context. Their influence on the context consists of modifying the component that is chosen as support. It is done using solids that are added or subtracted to the support. These solids are not complete components but rather parts of them. That is why they are not classified as auxiliary components.

Working of integrated components
As an integrated component is composed of a group of elements that are linked, it is important to define the relations existing between them and to integrate these relations into the behavior of the integrated component. The main components should be the master of the auxiliary components and the additional shapes, as the latter cannot exist if the main components do not exist. Therefore the auxiliary

component and the additional shapes have to position themselves relating to the main component and their dimensions have also to be linked to it.

The support
The support is simultaneously an input and an output of the insertion the integrated elements. Indeed the integrated component requires support for the positioning and, after the insertion, the geometry of the support is modified by the additional shapes.

Structure
Integrated components are inserted to complete the design of at least one component. That means that, after the insertion, the model contains at least two components and as a consequence the model has to be an assembly model.

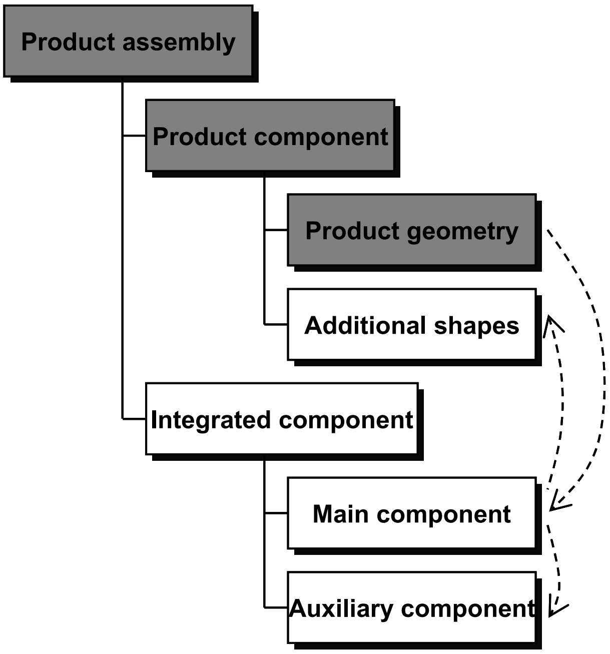


Figure 5-8: Simplified structure tree with an integrated component.

The structure of an assembly after the insertion of an integrated element is depicted in Figure 5-8. Elements represented in dark grey are those already present in the model before the insertion and the light colored elements are the newly added ones. This representation is only a generic example; in a concrete model, there could be many more components.

The integrated component is an assembly inserted in the product assembly. It has to be structured before the insertion with all the components it contains. The additional shapes have to be placed in the existing structure of the product.

The arrows show the links between the elements. They are oriented from the master element to its corresponding slave element. A link means that there is a relation between both elements and the slave element is depending on the master for:

- Its presence: The presence of the slave element is determined by the presence of the master since the slave cannot exist if the master does not exist. Second, an attribute of the master can determine the presence of the slave or its number. For example, a number of screws can depends on the dimension of the element they have to fix.

- Its positioning: The positioning of the slaves is directly dependent to the position of the master. The master is the element whose position has to be defined by users and the other elements have to adapt themselves. Thus all the elements of an integrated component can be moved by moving the main component.
- Its dimensions: In the same way as the position, the dimensions of the slave are contingent on the dimension of the master so that the entire assembly is changed by modifying the main component.

5.4 Skeleton

5.4.1 Requirements

The use of functional elements calls for new modeling methodologies; yet at this stage the modeling sequence is not defined. We have described three applications of functional elements - engineering templates, features, and integrated components - but the order they are to be employed in still remains to be examined. In addition, the first part of this document depicted interesting solutions based on skeletons (section 2.4.1.1).

From a general perspective, engineering templates define the global shapes of the product, integrated components define the secondary components, and features complete the design by defining the details. But these observations do not imply that the elements have to be inserted in a model in the order Engineering templates → Integrated components → Features. The reason why this simple sequence cannot absolutely be respected is that designers do not always possess the information they need, causing them to work in a way that can be considered as illogical. In fact, designers often need information from various sources and when they do not have all the information at their disposal, they have to start their work with incomplete information.

Therefore it is necessary to allow designers to build their model in the order they desire without altering the model quality. Hence, the relations between the functional elements have to be defined so that features, for example, do not necessarily need templates as support or so that a part of the product can be modeled without knowing the rest. *Thus it should be possible to create details although the basic shapes of the product are unknown.*

This way of working makes sense as, in some applications, the details are more important than the rest of the part. As a result, the main shapes of the product are not a support anymore but a wrapping for the details. For example, in the case of engines, the cylinder head has the function to surround the elements it contains and is not a basis for them. This means that it is easier to define the cylinders and all the channels and holes first and then to define the crankcase rather than vice versa. This methodology would be an improvement in comparison to a classical modeling method since it should match the development sequence of this product.

In conclusion, functional elements cannot be inserted one after the other just like a chain but should have an independent support. More arguments take the same direction and will be elaborated in the following.

Independently of the use of functional elements, there are several ways to position geometrical elements in a CAD model. Today's CAD systems offer the possibility to position an element relating to another by selecting it as support. For example, to model a hole, it is only necessary to choose the surface where it has to be positioned in order to insert it. This way of working is simpler than the use of coordinates. Yet it generates positioning relations between both elements in addition to the inevitable Boolean relation that is the basis of the solid modeling. While in the surface modeling domain, there is no Boolean relation, the same still applies for the positioning. Consequently there are elements in the model with two functions: the positioning reference and a part of the Boolean operation.

Even if this situation has advantages, it may provoke problems. The benefits are that the positioning of the elements is easier for the user and that a relative associativity is maintained, but this has several consequences:

- The first element must already exist to allow the modeling of the second one.
- The first element cannot be deleted without leading to problem by the definition of the second one.
- Some modification of the first element can make the relation with the second one inconsistent.
- A modification in the model structure can also lead to a consistency problem.

In summary, associativity imposes a given order in the modeling sequence and can result in update problems. Moreover, it is possible to create cross-relations so that an element is not necessarily positioned on the element to which it has a Boolean relation. In such a case, the multiple relations of any kind make it hard to understand or to modify the model structure or to delete elements. Finally, the fact that geometrical elements can be used for the definition of the solid and for the positioning of other element leads to an ambiguity about their function. It would hence be interesting for both functions to be supported by different elements.

5.4.2 Skeleton as Support for Functional Elements

To solve the problems described above, a skeleton will be introduced in the model. We will see in this section that the combination of skeleton and functional elements can yield significant advantages. In anatomy, the skeleton is the stiff supporting framework of the body. Analogously, we will define the skeleton as follows:

Definition 5-6: Skeleton
The skeleton is the outline structure providing support to the geometrical product model.

Figure 5-9 shows two pictures of the same component modeled with and without skeleton. The broken lines show the positioning relations between two elements. In the first picture, the geometrical elements are linked by means of positioning relations and geometrical operations whereas, in the second picture, a skeleton has been added to serve as global support for the element positioning. In this way the cross-relations are suppressed and more flexibility is gained.

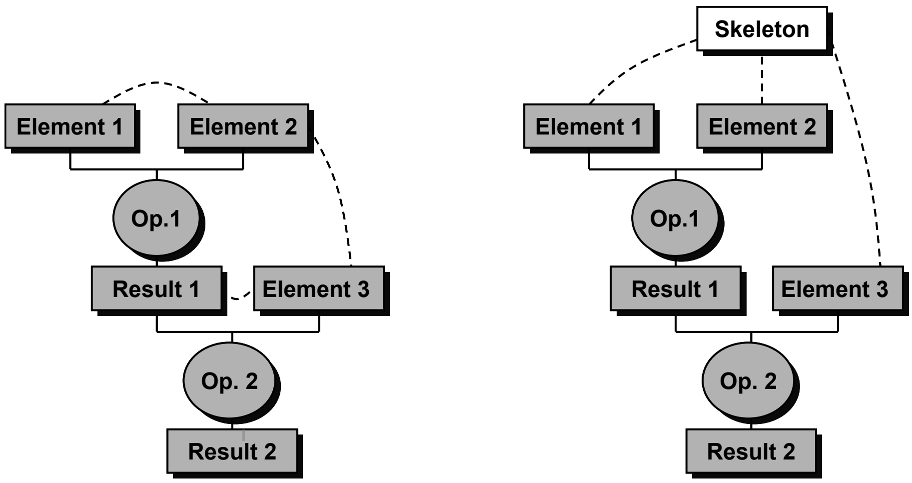


Figure 5-9: Advantage of a product skeleton.

Clearly, modeling with a skeleton introduces a new task: that of modeling the skeleton before modeling the geometry. Yet this solution is not thought to work in a usual modeling methodology. The skeleton will be employed in combination with functional elements – and, first of all, with engineering templates – so that a solution can be found to avoid additional effort for the user.

The picture on the left in Figure 5-10 illustrates the classical modeling methodology where the geometry is directly based on the reference system and positioned by means of coordinates and measurements. The picture on the right represents the new situation with the skeleton between the geometry and the component reference system. The advantages of such a solution are taken from the fact that knowledge can be added to the skeleton.

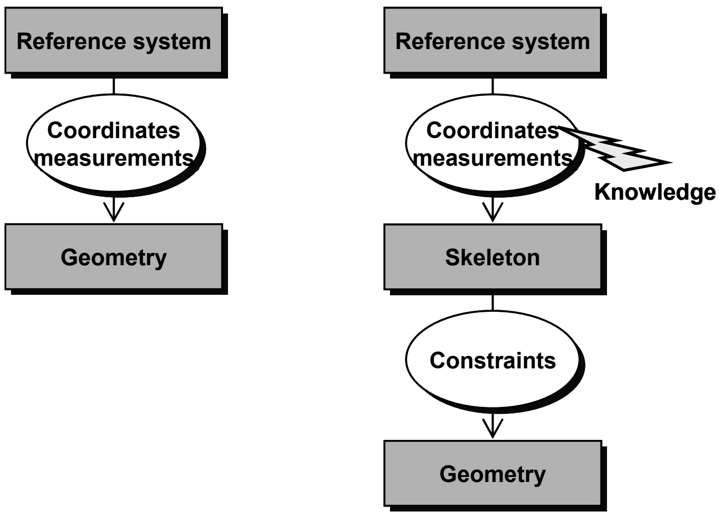


Figure 5-10: Skeleton working

5.4.3 Technical Solution for the Skeleton

While the objective of the skeleton is to position the geometry, it can also determine some of its dimensions. Indeed dependencies between the elements make the position of some elements dependent on the dimensions of others, or vice versa.

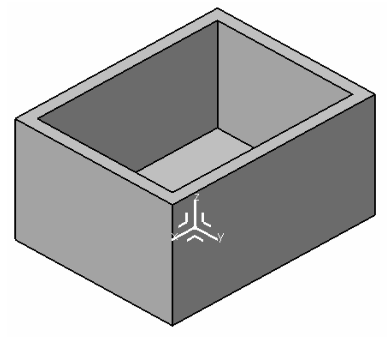


Figure 5-11: Example of a box.

Let us look at the simple example of a box (see Figure 5-11). For the user, this object has three main dimensions: the length, the width, and the depth. But for designers additional information must be considered in the form of the reference system. We will consider that the reference system is placed in the middle of the bottom of the box and that the thickness of the box is not important for our demonstration. There are two alternatives for designers to create this model:

- Model the bottom and then the side faces independently. In this case, the side faces need to be positioned so that the distance to the center is the half of the bottom dimensions (Figure 5-12).
- Apply constraints to make the side faces coincide with the bottom contour (Figure 5-13).

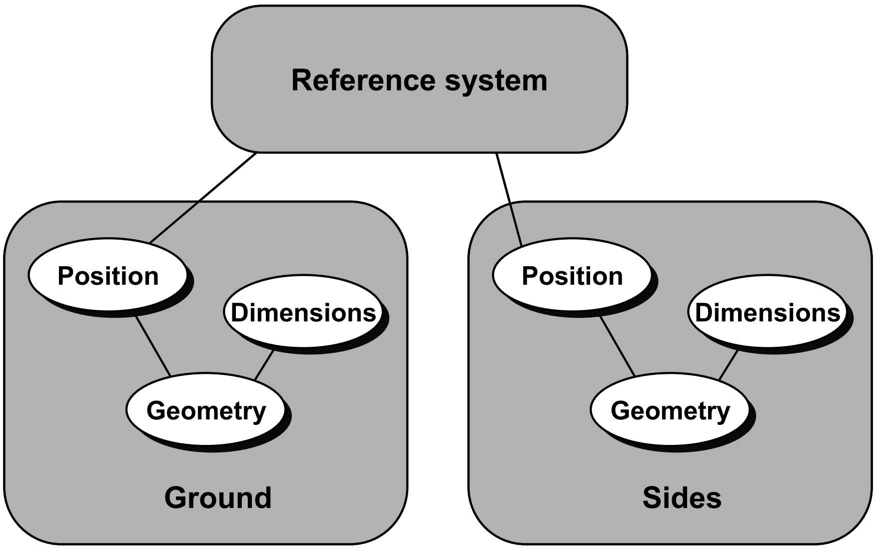


Figure 5-12: Use of reference system for the positioning.

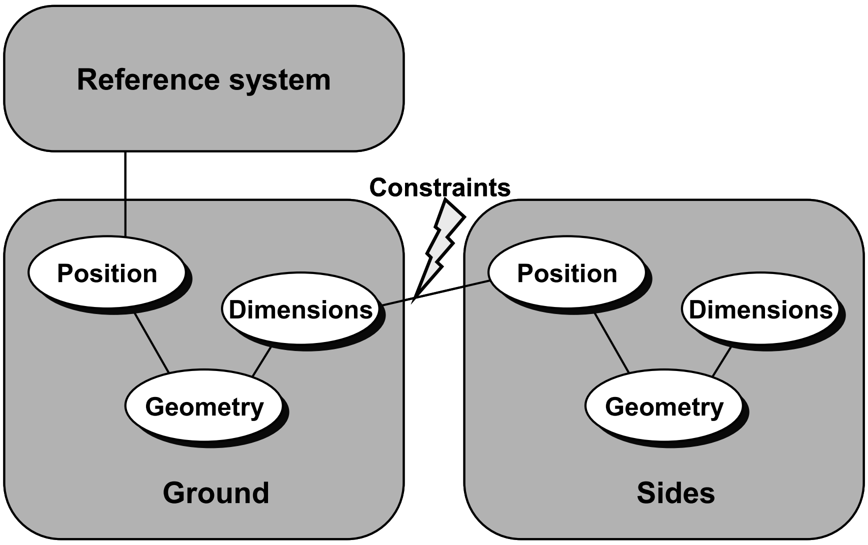


Figure 5-13: Use of constraints for the positioning.

The second solution shows the dependency between the dimensions of one element and the position of other ones and it also shows a short application of knowledge. As both the dimensions of the ground and the position of the side faces are linked, we can imagine that only one element of the skeleton adopts both functions.

Nature of the skeleton

The question is then to define what the skeleton looks like in practice. It is composed of wireframe geometry such as planes, lines, or points. The choice between these three possibilities depends on the applications. Generally points are used for elements having a symmetry center, lines for elements having a symmetry axis, and planes for the rest. The advantage of planes is that they can be used at the same time as positioning elements and as supports for profiles. Thus the creation of the skeleton has to be done by an analysis of the design and its dependencies.

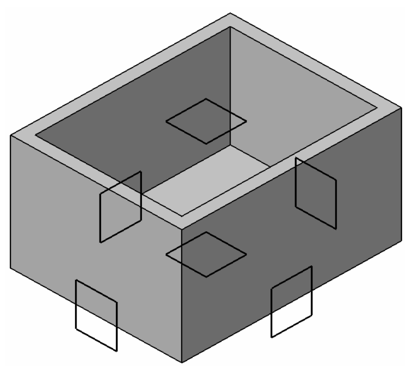


Figure 5-14: Use of skeleton for the positioning.

In the example of the box, the skeleton would be composed of six planes (Figure 5-14) delimiting the six faces. The side faces could have been constrained with lines but, as mentioned, the planes offer more advantages for further detailing of the model.

Insertion of knowledge

The strong point of the skeleton is not seen in small models but rather in complex ones as it enables references to be generated only once for several elements, and *the insertion of knowledge is easier.* In the case where a model is made without a skeleton, the knowledge has to relate directly to the geometry and the difficulties appear when the dimension designers wish to use is not explicitly available. This occurs when the dimension results from the combination of two or more other dimensions.

Figure 5-15 sets out an example where the component is composed of a table and a block. Each is the result of an extruded profile (thick lines in the figure). The dimensions of the profiles must be given to generate them; yet if the designer is interested in the flange width (for example, to keep it constant), it must be calculated from the difference between the table and the block dimensions. This operation is not difficult but we have to consider that there are, in a real model, many more constraints to define. Consequently, the creation of all the rules would be time consuming and would complicate the model.

Another solution is to constrain the table profile so that the parameter between the side faces of the block and the table is constant. This solution has the disadvantage that it refers to profile edges whose definition can change during a modification. That means that a modification of the profile forms would lead to update problems.

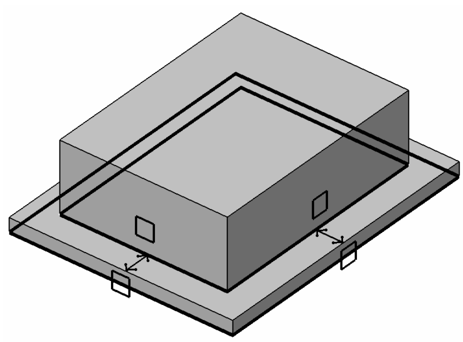


Figure 5-15: Combination of skeleton and knowledge.

The skeleton is a good methodology to deal with the difficulties mentioned. *The solution is to base all the knowledge not on the geometry but on the skeleton. This makes it much easier to have a clear view of the geometrical dependencies.* Moreover, adding knowledge in the model becomes more comfortable as the rules will refer to wireframe geometry and not to edges or faces, which are non-associative elements.

In the given example, the solution would be to create a skeleton containing planes for the limiting of the geometry. The planes for the table would be directly defined on the basis of the planes for the block with a constant distance. For this reason, a move of the block planes automatically involves a move of the table planes.

Consequences of the skeleton utilization
The utilization of skeleton and knowledge improves the legibility of the structure tree and allows a better comprehension of the dependencies in models. However, the graphic field representing the geometry of the product in the CAD system contains more information because of the skeleton. Designers will then have to work in a new way:

- By defining the dimensions of profiles or solids, the geometry of the skeleton has to be selected as reference instead of giving dimension values.

- The positioning of elements is done using the reference of the skeleton instead of selecting faces of the existing geometry.
- The modification of the model dimensions is done by manipulating values of the skeleton parameters instead of changing the geometry.

Standardization and reusability are also important aspects of working with skeletons. In fact, a skeleton can be used as the first step in modeling a new project. This means that a standard skeleton has to be available for a product family and has to be adapted to the new requirements before creating or inserting the geometry.

5.5 Synthesis About the Utilization of Functional Elements and Skeleton

The three kinds of functional elements and the skeleton are a set of methods aiming to describe a product. Yet, which of them is to be applied in which situation or for which application? And how do they interact with each other? This section is intended as the synthesis of previous sections and presents the description of a global modeling methodology.

Relation features/skeleton
Because of the standards and their simple form, features are typically elements whose dimensions are relatively fixed or have a limited range. And the fact that they are used to define details of the product means that they are inserted on the basis of its main shapes. As these main shapes are delimited by the wireframe geometry of the skeleton, features have to be developed so that they can be inserted on the skeleton and not directly on geometrical elements. This solution also has the advantage that features can be inserted before the main shapes of the product are defined.

In summary, the relation between features and skeleton is the following: the skeleton has the function to position the features. It can eventually be used to define their dimensions but, most of the time, the dimensions will be defined using the feature's internal parameters.

Relation engineering template/skeleton
The skeleton is useful to position the engineering template but also to determine its dimensions. In fact, the skeleton is a means to adapt engineering templates to the product requirements and, for the rest of the thesis, we will not dissociate the engineering template and skeleton use.

Relation-integrated element/skeleton
The influence of the skeleton on integrated elements has to be placed between features and templates since integrated elements are generally standardized elements as are features but they can be as complex as templates. Clearly, the skeleton should serve as positioning element (for the same reasons as for the features) but the decision as to whether or not it also has to define its dimensions depends on its standardization degree. Provided that the functional elements are strongly standardized, the skeleton will not have the function to set its dimensions. But if the functional element dimensions depend on the context, the skeleton will be used to define them.

Modeling sequence and product structure
The skeleton offers the greatest benefits when it is employed as a reusable object since it would be too expensive for designers to create a new skeleton at the beginning of each new model. Further, if a product can be modeled as a reusable element (that is, a template), it is logical that the corresponding skeleton is reused as well.

The question is then whether skeletons have to be inserted into templates or if they have to be independent of them. On principle, both alternatives are possible but they affect the working methodology. It depends on the way the product has to be modeled. In the next sections we describe both options.

Independent skeleton
The solution consisting of working with a skeleton that is independent of the geometry (and particularly of the templates) is schematized in Figure 5-16. The skeleton, the templates, and the elements of the detail design are represented independently. The solid lines represent positioning relations, whereas the broken lines represent other kinds of link such as knowledge dependencies. The dark grey rectangles group the elements of the same overall component or assembly.

If features are common to all the items of a product family, they can be automatically inserted in the template. For this reason, the template already contains features in this representation.

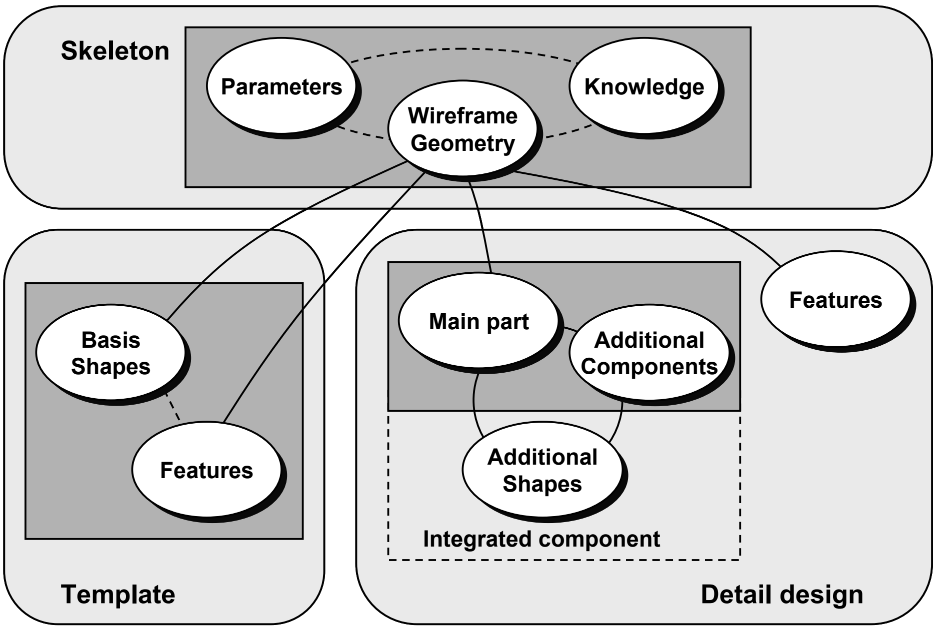


Figure 5-16: Independent skeleton.

In this case, the skeleton is inserted into the model before the geometry. Its parameterization and knowledge allow designers to prepare it by assigning it the right parameter values. Consequently, all the reference elements of the product are available to start the insertion of the functional elements in the order the user desires: it is possible to insert features or integrated components before engineering templates.

Skeleton integrated into engineering templates
A product skeleton is the support for the corresponding engineering template(s) so that they are strongly linked together. It is thus conceivable to integrate the skeleton in the engineering template. With this solution, the insertion of the engineering template is easier since the links with the skeleton are already insured. As for the first solution, integrated components and features are inserted with the skeleton as reference. Yet, they can no longer be inserted before the template, as the skeleton is not available separately. The links between the elements and their arrangement are depicted in Figure 5-17.

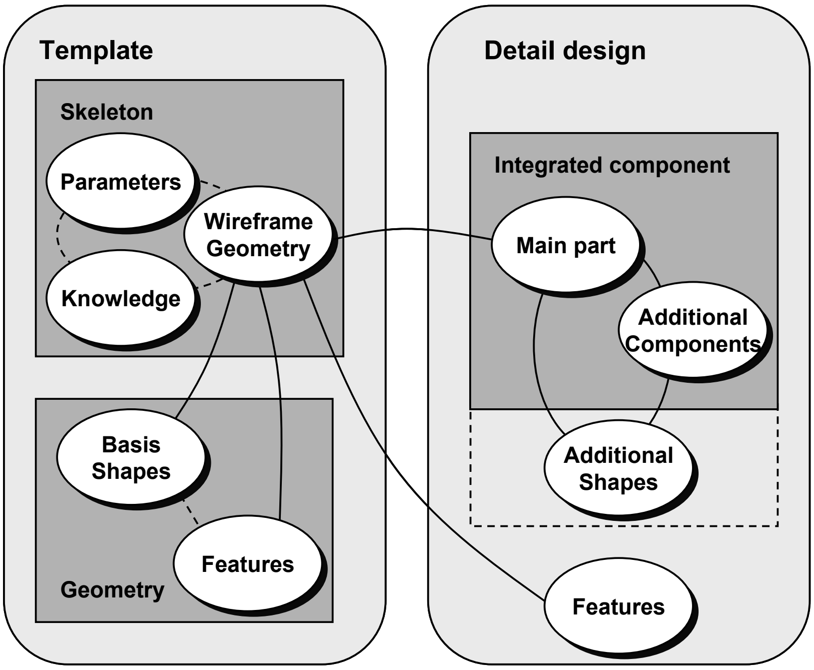


Figure 5-17: Integrated skeleton.

It is possible to combine both solutions so that there is a part of the skeleton that is independent of the rest of the geometry and a part of the skeleton that belongs to templates. The details represented by integrated components and features are then inserted either on the main skeleton or on the template skeleton. Figure 5-18 shows the simplified relation network in the product model where there are several skeletons. This configuration is particularly interesting if the product model is made up of several templates.

Template skeletons have to contain only the reference elements necessary for the corresponding templates. In the main skeleton, there are the reference elements of a higher level having the function to manage the complete geometry. In particular, the main skeleton insures the consistency of the model and defines the relative position between the different templates.

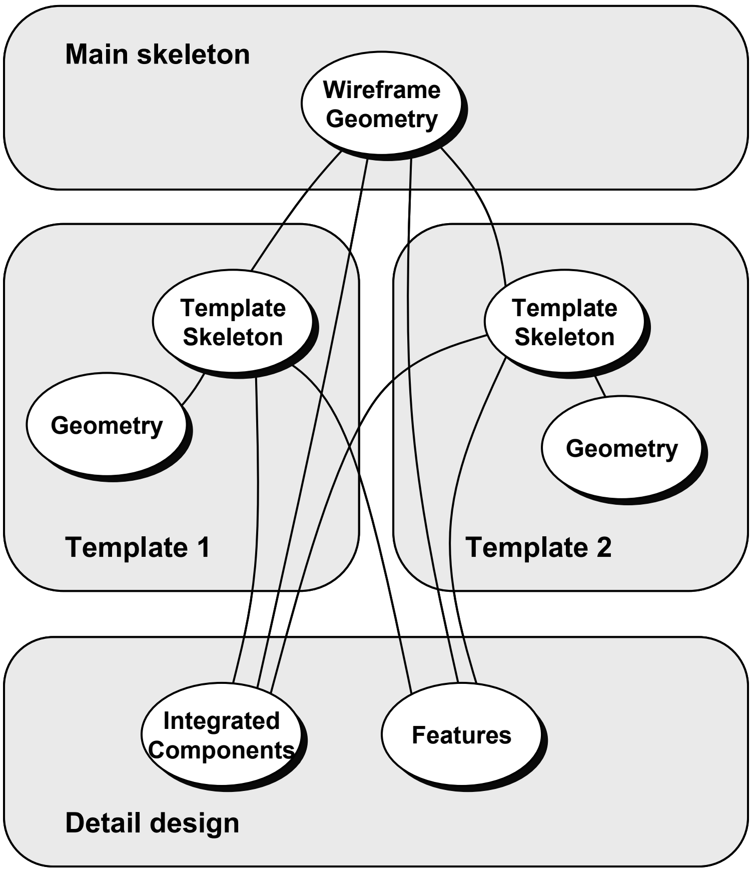


Figure 5-18: Combination of integrated and independent skeleton.

5.6 Concurrent Engineering

5.6.1 Introduction

It is well known that the product's time to market is to be reduced and, at the same time, the quality has to be enhanced. Thus a new product design methodology cannot be validated if these criteria are not met. We will see in this section how functional elements contribute to the satisfaction of these requirements by supporting concurrent and collaborative engineering.

Time to market
The time to market of a product depends on the product definition and the production time. In line with our topic, we will concentrate on the design time. This is the sum of the times of each step of the development process. Therefore, to reduce the global design time, there are three main directions to follow:

- *Suppress superfluous steps*. The analysis of the overall process chain can detect if some steps are unnecessary so that they can be suppressed. Unnecessary steps in the process chain are due either to the organization that is not optimally realized or to the fact that the previous step does not deliver an optimal result. For example, there are interface or quality problems when data are transferred from one application to another. As a consequence, data have to be processed additionally or corrected to conform to expectations. Such a task could be avoided by a global view of the systems and the data flow to optimize or suppress the data conversion and perhaps change the process sequence.

- *Reduce the duration of each step*. To reduce the global process time it is clear that each step has to be improved. For that, the inputs and the outputs of the step have to be correctly defined as well as the tools and methodologies related to it
- *Make each step start earlier*. Whatever the duration of the steps is, the global process time can be reduced if each step commences earlier. That means that a sequential organization has to be changed in a concurrent one. The consequence is that a step should start, if possible, before the previous one is achieved.

Early availability of relevant information
To make it possible to start a step before the previous one is achieved, it is important to define properly the inputs and outputs of each step. Traditionally the outputs of one step are the input for the next one. That means that the aim of a step is to prepare the information necessary for the next step.

In order to implement a concurrent engineering environment, it is necessary to consider the notion of time. Thus the question is no longer "What does the next step need?" but rather "What does the next step need at which time?" However, in order to be able to start without having a final data version, certain conditions are needed. As elaborated below, two cases can be identified.

The first case is the one where only a part of the outputs is required to start a following step. An example of such a situation is given in Figure 5-19. The process consists of three steps. The results of the first one are the inputs for the second and third steps.

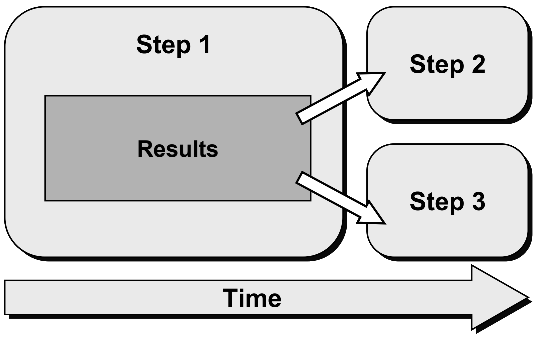


Figure 5-19: Sequential organization.

In this situation, an examination has to be performed to find out what is really needed for the following steps. If, for example, Step 2 requires only a part of the result of Step 1, then it can be started earlier than Step 3 so that it will be finished earlier as well (see Figure 5-20). It is important to note that Result 1 in the figure is not an intermediary state of Result 2 but a separate result. That means that Step 1 is a task that delivers several differentiable results.

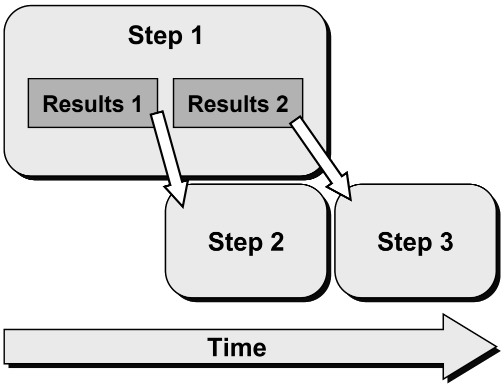


Figure 5-20: Parallel organization.

In contrast, if the last result is needed to start a step (see Step 3), it is harder to begin the other task before its achievement. To deal with this, another solution is proposed in the next section. However the subdivision and the organization in Step 1 are clearly important: a given sequence must be respected.

The second case is when the result of a step is the mandatory prerequisite for a subsequent step. The subdivision into different results is not a solution for this problem but the reflection has to be oriented in the direction of the intermediary states of the final result. In fact, it is often possible to start a task with a data version that has not been finalized.

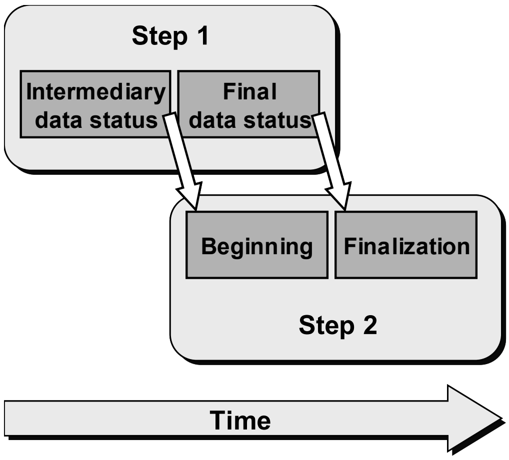


Figure 5-21: Work with intermediary data status.

Figure 5-21 gives a principle representation of two tasks that can be parallelized. In this representation the final data stand is of the same type as the intermediary data stand but in a finalized state. This has various consequences:

- Although the intermediary data status does not yield a complete representation of the result, it has to be precise enough to allow the start of the second step. Although this requirement shows that a methodology must be developed and respected to insure that the process works well.
- A methodology is also necessary for the second step so that it can work even if an incomplete data set is used.
- The system should make it possible to update the inputs of Step 2 as soon as Step 1 is finalized. That means that systems supporting associativity methods

are needed as well as communication bridges between the systems when they are different.

Requirements to enhance the model and product quality
In addition to the early availability of necessary data, more aspects are important to improve the product quality. Because of the deadlines imposed on the product development process, product quality depends among other things on the efficiency of the design and thus on the model quality.

To improve the model quality, the following aspects have to be considered:

- Avoiding errors. Thanks to an associative data and information flow, all the views of an element in the process chain should be up-to-date so that errors can be avoided.
- Avoiding improvement loops. Improvement loops should be avoided by setting up a system of prioritization that should give each designer the possibility to anticipate the further steps (see also section 4.4.1).
- Avoiding repetitive work or remodeling. Repetitive work should be suppressed by employing reusable elements.

5.6.2 Contribution of Functional Elements to Collaborative Engineering

Differentiation between main shapes and details
The requirements of section 5.6.1 mention the fact that a process chain step could begin with an unfinished data set. If so, the available data have to give a first impression of the model so that designers know in which direction they have to work. This means that they need the main characteristics of the product and not necessarily the details. That why the differentiation between the functional elements makes sense: in this context, templates represent the main shapes of a product and it is not necessary to have defined all the details with the help of features before starting subsequent steps.

Templates have an additional advantage in comparison to the traditional modeling methodology: habitually designers do not model an element as long as it is not fully defined. On the contrary, the use of templates makes information available although designers have not yet thought about it. This information is present in the template and, while it has not yet been adapted to the current project, it may be employed by the following step. And as soon as designers have set it, it is updated in the following step as well.

Reusability
Reusing an element has two advantages: first it saves time by suppressing repetitive work, second it allows a following process to start earlier thanks to existing information. In fact, engineering templates should be a support for each step of the design stage. A template is the synthesis of the results of the models of the previous products of the same family. Thus by using a template within one step, the result of the same step of the previous serials is utilized. However the template cannot be employed as long as it does not have the necessary inputs. That is why the inputs of the previous step of the previous serial could also be used as a starting basis.

This principle is illustrated in Figure 5-22, where the considered step is Step X from Serial Y. This step could benefit from the same step of the previous serial (Step X, Serial Y-1) as well as from the previous step of the previous serial (Step X-1, Serial Y-1). In this context, the results of Step X-1 of the same serial are not necessary to start step X.

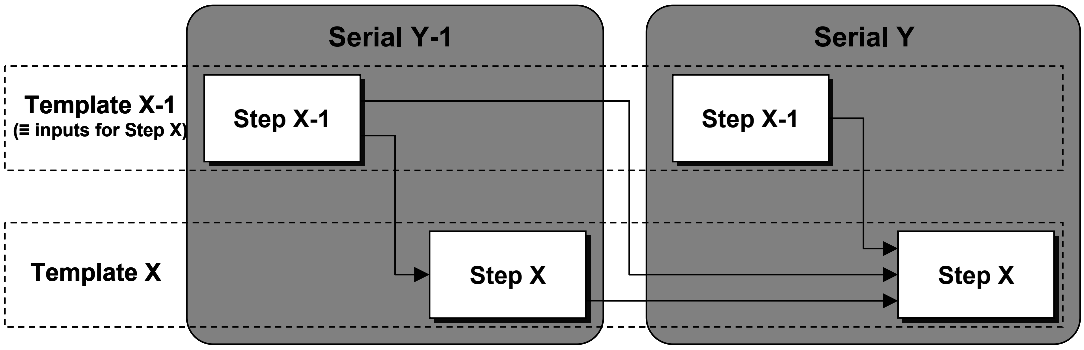


Figure 5-22: Use of previous results as engineering template inputs.

Thus, deploying the engineering template methodology, it would be possible to start all the process activities not only before the previous ones are achieved but also before they have delivered any information. Indeed all the activities could start at the same time.

As a result, this method is a contribution to the development of a process foreseen by C. Bouchard and A. Aoussat, who describe the future car body process as shown in Figure 5-23 [BoAo-03].

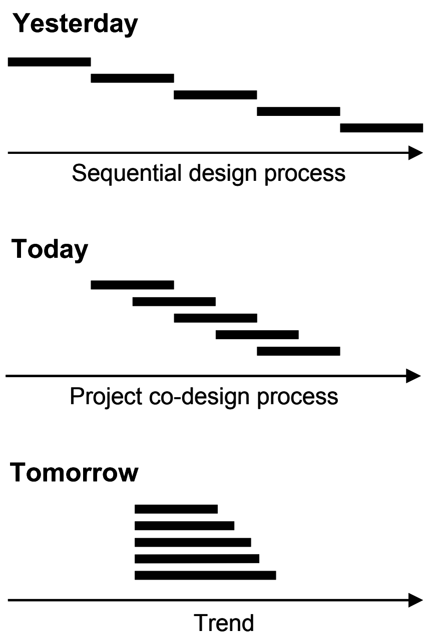


Figure 5-23: Parallelization of processes (based on [BoAo-03]).

Precise working methodology and sequence

As functional elements and skeleton make it possible to work in the desired order, a modeling sequence can and should be defined to make the process work as efficiently as possible. This means that the task priority of each step must be defined to deliver the result at the right time. Thus the process chain cannot be seen as the sequencing of various steps that would be represented as black boxes. The optimization of the process chain and the optimization of each step cannot be done independently.

Structured information model
Functional elements have a standard structure so that the model structure tree achieves enhanced legibility. But the allocation of each element and their semantics has to be defined to facilitate access to the required information.

Associativity
The aim of the associativity is to create and maintain a link between two elements so that modifications of the first element are reflected in the second one. This method could be applied to support the scenario of Figure 5-21. In this way, Step 2 always operates on the same data but with various definition versions. That means that the data from Step 1 are used for further applications and, when the final version is ready, the intermediary version will not be replaced with the new one but simply refreshed.

The risk is that the development carried out in Step 2 will no longer suit the new version of Step 1. That is why both the first and second steps have to foresee the evolution of the first-step results. An intermediary version can only be delivered if it represents a consistent view of the data in comparison to the future result. Additionally, in the second step, it is essential that the development can fit any possible evolution of Step 1.

The associativity mechanisms can link two or more elements of the same model, different models of the same application, or different models of different applications. Associativity is used to maintain the consistency of an element in all the steps of the process chain. In fact, an element may have various views in the design process, all of which have to match each other. The ability to guarantee this associativity lies in the software functionalities and should, above all, be supported by the PDM system. More details about this subject are given in the next section.

5.6.3 Product Data Management

5.6.3.1 PDM Systems and Functional Elements

As with CAD systems earlier, PDM systems have been recognized as a central point of the product development process. They represent a platform for the integration of a wide variety of systems aimed at product description (geometrical or non-geometrical). The aim of this section is not to provide an exhaustive description of the PDM system but to define how functional elements and PDM systems should interact in a productive situation or how PDM systems should support the modeling process first for the definition of functional elements and second for their utilization. Each of the three functional elements and the product skeleton have different impacts on working with a PDM system as they may or may not affect the product structure.

Engineering templates and skeleton
In PDM systems, there is the possibility to create a product master and product variants. The product master is a model containing the information common to all the products of a product family, whereas the variants are derived from the previous one and contain the information specific to each product. *Consequently it seems logical to*

establish engineering templates as product masters and the derived models as product variants. But the question could be more complex if templates are not the basis of a new model. In fact, in the context of working with PDM systems and templates, the derivation of a product variant will be the first step in the modeling of a new product. Thus we need to define which information is available at the beginning of the project.

We have mentioned that a model can be detailed without having its basis shapes represented by a template. This is possible by using a product skeleton that is external to the template. In this case, the master should be the skeleton model and the templates will have to be inserted later. However, templates and their corresponding skeletons generally should not be independent so that product masters would be composed of template/skeleton couples.

Integrated components
Integrated elements should be available for all users in a component library. This requirement can be easily met by PDM systems. However, a more important issue is to define how integrated components should be inserted in the model since, as soon as a PDM system is employed, there are several alternatives available. Inserting components in a model means defining the product configuration. Typically, this task is done in the PDM system. However integrated components are more than standards components since they also modify the geometry present in the model. At present, the insertion of integrated components can only be done in CAD systems as only these systems support this particular methodology. After the insertion of integrated components, the structure product is modified and has to be reflected the in the PDM database.

Functional features
The use of functional features is quite simple as it does not change the product structure on the assembly level: it only modifies the geometry of existing parts. The insertion of functional features is done in the CAD system by means of libraries saved in the PDM system.

5.6.3.2 Creation of Product Sub-families in a PDM System

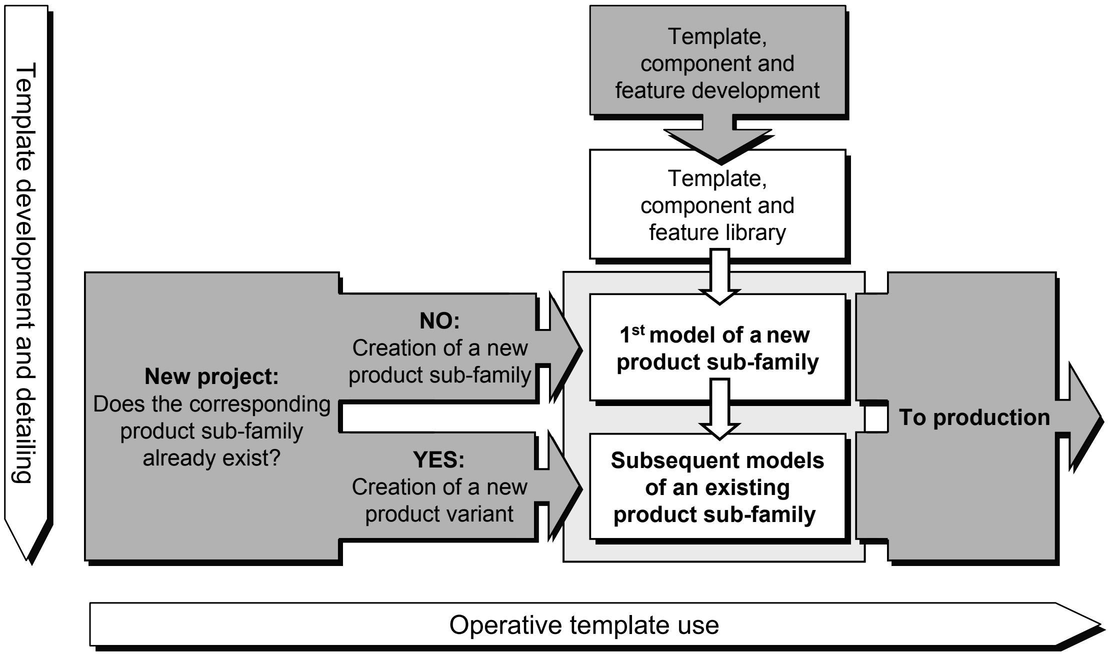


Figure 5-24: Working of functional elements and PDM system.

Figure 5-24 shows the working of functional elements and PDM systems. There are two axes in this representation corresponding to the template development (vertical axis) and the template utilization (horizontal axis). The first task is the creation of a template, component and feature library. Then follows the functional elements use: at the beginning of a new project the question should be asked if there is already a product of the same type:

- If there is no product of the same type, a new one has to be made on the basis of the functional element libraries. In this way, a new model for the production is created but, at the same time, a new product sub-family is generated. This means that the model that has just been created represents a new product family in the form a detailed template.
- If there is already a product of the same type, it will be used to directly generate the new model. Then the task is not to detail a standard template but to reuse a model and to adapt it.

As a result, the original template library is permanently used to create a larger library of models that are more detailed. Templates cover the needs of a product family and detailed models are more specific. That is why we talk about a "sub-family". Product sub-families are not developed immediately owing to the enormous number of configuration variants. Creating such a large number of sub-families would generate more work than directly modeling the product with the traditional method. Yet, this is not sensible as templates aim at reducing design time. This process enables the size of the library to be increased without additional work since finished models are simultaneously sub-family templates. Moreover as sub-family templates are based on family templates, they automatically integrate all the advantages represented by the structure, the knowledge integration, process information, etc. Finally this methodology is supported by PDM systems in the sense that they make it possible to generate product variants from a master model, as explained in the next section.

5.6.3.3 Product Versions

An example of functional element usage with help of a PDM system is set out in Figure 5-25. On the left, the generic structure of a product model is depicted. It is the template model with some components and some empty containers for the insertion of further elements. These are present in the model to structure it properly although not all of them have as yet been defined. Empty containers concern either whole components or a part of a component's geometry. In the figure below, they are represented by rectangles with broken lines.

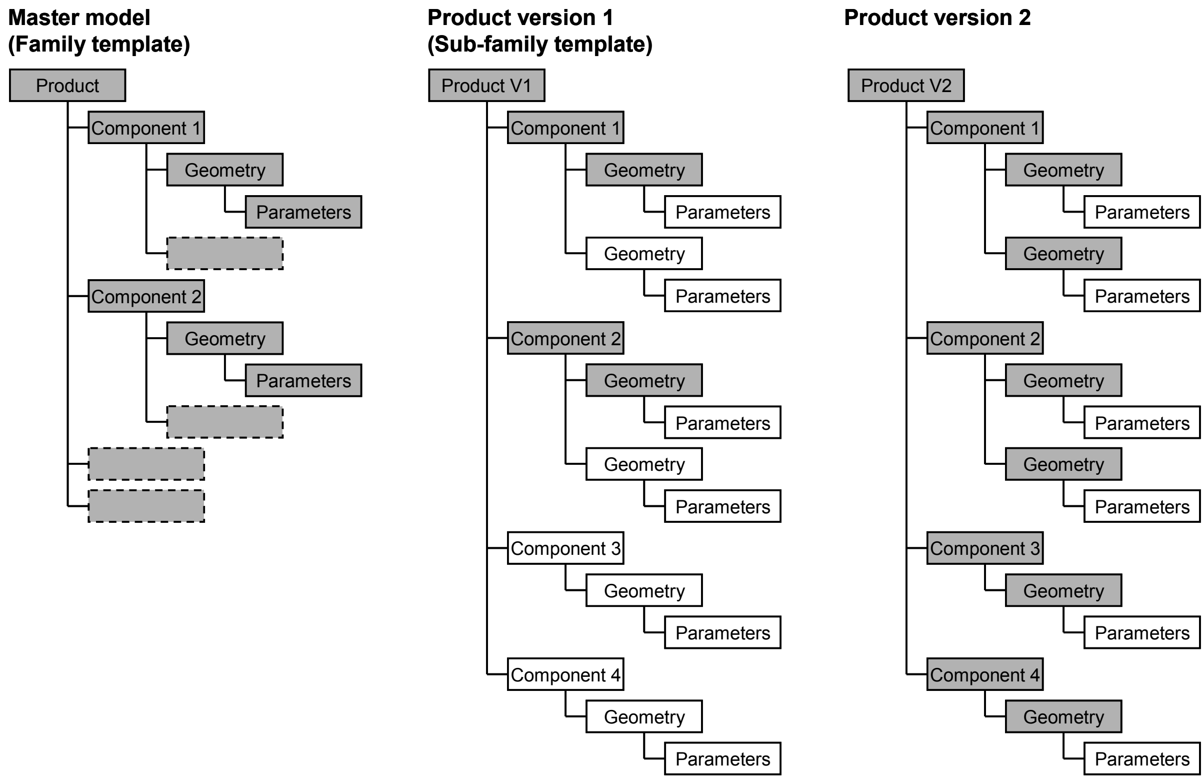


Figure 5-25: Derivation of product variants from a product master.

A second time, the master model is used to create a new model. This is represented in the middle of the figure. The rectangles colored in gray stand for unchanged elements whereas those in white identify new or modified elements. It can be seen that the existing geometry has been modified by modifying parameter values and that empty containers have been filled.

On the right-hand side of the figure, the structure of a product that corresponds to an existing sub-family is displayed. It is derived not from the master model structure but from the sub-family structure represented by the previously modeled product (in the middle of the picture). In the same way as earlier, the modified elements are represented in white. Here, only parameters values have been modified as the geometry of the sub-template is identical to the desired one. The basic geometry that is common to all the models is not saved several times in the database since, although the second variant is derived from the first one, it still has a link to the master model.

5.7 Conclusion

This chapter has introduced three applications of functional elements that should be used together with a skeleton to improve the design process. There are three kinds of objects employed at different levels of the product models. Whereas engineering templates are the basis of the models, integrated components represent sub-assemblies of the models, and functional features achieve their detailing. The benefit generated by this new modeling method is particularly clear from the point of view of the process chain and in the case where various models are linked together. Indeed functional elements are an essential support of concurrent engineering (through their reusability) and of collaborative engineering (through their ability to provide information from other steps of the process chain).

Methodologies, technologies, model structures and tools are strongly linked together. It is not possible to introduce a new methodology such as the one based on functional elements without having a general idea of all the consequences affecting the designer's work.

With the utilization of functional elements and the concurrent engineering scheme, not only are the results themselves important but also the way they are obtained. Thus more and more constraints are imposed on designers, who have to adapt their working methodology and habits. As a consequence, the ideal process that can be developed in such a thesis has to be confronted with reality; above all, the issues of user bias and user qualification must be taken into account as well as all the constraints generated by the company's history and specificities.

The next chapter will be devoted to the description of a concrete example of the implementation of the methodologies developed until this point of the work.

PART III
Application

Chapter 6
Component and Die Face Engineering Templates

6.1 Introduction

In the first part of the thesis, the current situation in the stamping tool design process was described; in the second part, a concept for engineering templates, integrated components, and feature use was developed. This third part of the thesis aims at demonstrating the benefit of the methods described in the last part when applied to stamping tool design. To do so, a car body component has been chosen and used for the illustration of the overall process chain of interest: the hood.

From the modeling perspective, there is a natural separation of the stamping tool design process into two parts. In fact, the beginning of the stamping tool development process is a part of the component definition as the activities carried out in this domain look downwards. They concern the definition of the component geometry and the die face across the method planning and the simulation. This deals with surface modeling and the definition of elements that are directly dependent on the component geometry. The second part concerns the modeling of the stamping tool as solid and the subsequent steps casting and machining. This part looks upwards as the definition of the stamping tool as hardware does not really influence the component definition but is rather oriented toward production.

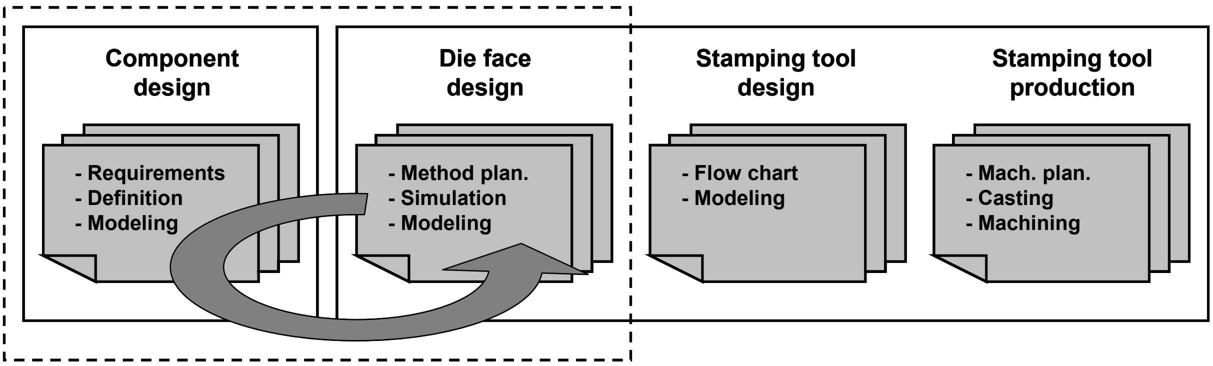


Figure 6-1: Separation in the development process.

In practice, nevertheless, there is a separation between the component design and the die face design as the organization has integrated the die face design in the stamping tool definition process. Figure 6-1 shows the arrangement of the activities by regrouping those that belong to the same organization in the same rectangular frame. However the arrow indicates that there are iterations between the die face design and the component design (particularly because of the simulation that validates the component geometry), so that we can consider both activities as belonging to the same sub-process.

For these reasons, the present part of the thesis has been divided into two chapters: a common definition of the component and die face modeling and a definition of the stamping tool with a look at the downstream processes. Then the third chapter confronts the methods defined so far with the conditions of a real application.

The use of engineering templates for the die face cannot be optimal if no engineering template is employed for the component as well since the die face shapes are directly derived from the component shapes. That is why the component modeling methodology is described in addition to the die face modeling methodology. This chapter deals with the definition of the two templates, which are both surface templates. It introduces a solution to create and use the templates. Both the integration of the designer's knowledge and the simulation information are clarified and an application scenario using the hood as example is depicted.

6.2 Engineering Template Development Method for Car-Body Components and Die Faces

6.2.1 Introduction of the Method

First of all, the analysis of several car body components and their corresponding die faces shows that it is not possible to make one generic template for all the components. It is clear that the component and die face geometry are too different from one type to another, so that one template per component and die face type is needed. Thus, in our case, a template has been developed for the hood and cannot be used for other types of parts. However, what is interesting about this template is that it can be used for several car models or variants.

The creation of the template is based on three stages as shown in Figure 6-2:

- Information acquisition
- Information analysis
- Template modeling

These points are developed in the next sections.

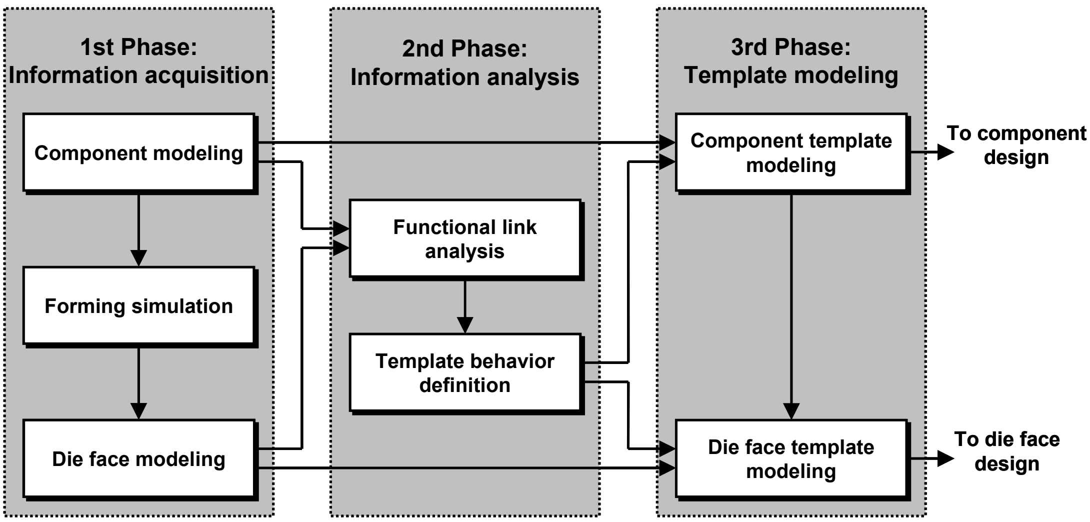


Figure 6-2: Engineering template development process.

6.2.2 Information Acquisition

To develop templates, it is paramount to analyze existing designs. The interconnection mentioned above, between the component and the die face, necessitates that both respective templates have to be developed at the same time.

Thus information acquisition consists of initially carrying out the process of the component modeling whose templates are to be created to achieve reference models. If necessary, not only one component may be analyzed but several ones of the same type in order to identify their similarities. All the activities described in chapter 3 up to the die face modeling have to be performed: component modeling, method planning, simulation, and die face modeling.

The result of this sequence is three models: component model, simulation model, and die face model. The die face model is created on the basis of the simulation model. Thus this model contains process information in the form of geometry validated by the simulation. But there is no process information in the component model and there is no link between the models.

6.2.3 Information Analysis

The information analysis stage consists of comparing the component and die face models to identify the functional link between them and to deduce the behavior of the template to be created. Functional link analysis means identifying the geometrical elements that are present in one or the other model and also identifying the elements of one model that have an influence on elements of the other one.

Then, on the basis of the functional link analysis, the template behavior has to be defined. In other words, following points have to be treated:

- What should each template contain?
- How should the templates be structured?
- What should be the relationships between the elements within a template?
- What should be the relationships between the elements of both templates?
- What should be the control parameters?
- What are the rules to be implemented?

6.2.4 Template Definition

The models created in the first stages are not linked, and their structure and behavior do not meet the requirements of the template methodology. For this reason, the third stage consists of the concrete modeling of the template and the implementation of the behavior described in the prior stage. The resulting geometry resembles the original geometry except for the fact that some details that are too specific are absent in order to make the template as general as possible.

Yet, as in the previous stage, which dealt with the ideal template behavior, the implementation is dependent on the software features. The result of this stage is two templates (a component and its corresponding die face), which will then be used for the design activities.

6.3 Application of the Component and Die Face Template Development Method

6.3.1 Model Analysis

6.3.1.1 Component Model Analysis

The models to analyze here result from the design of the component made independently of the future modeling process. This means that downstream processes are only envisaged in the sense of feasibility, cost, etc. of the component but not in the sense of model creation. In today's process, work has been done to define a standardized modeling methodology. In this framework, the solutions found aim at optimizing the component design stage itself and this with two focal points:

- First modeling of the component
- Modification of the component model

Consequently the standardization of models targets defining where each element has to be placed in the model structure and in which order they have to be created. In this way, the intervention of various actors in the same project is simplified. Moreover, the optimal modeling method in the view of the stability of the model during modification is established for all the designers as well as the logical links between elements within the model to define the repercussions of an element modification.

As cited above, subsequent processes have not really been envisaged in the modeling method of car-body components except for the case of holes. Holes have been a matter of particular attention as they have a preponderant importance in the overall design and manufacturing processes. Holes features are currently used on a widespread basis in car-body design. They contain information needed by the inspection planning, for example, the coordinates of their center-points and the orientation of their punch direction. As the focus of this thesis is the stamping tool design process, we will not further detail the properties of this kind of feature.

Other features are also in use in the car body design process. They are form features such as beads, pocket, or flanges. However, in the same way, they have been thought to support the design modeling stage and do not consider downstream processes.

6.3.1.2 Die Face Model Analysis

The work of die face designers bases directly on the component geometry. This means that a copy of the finished component geometry is used. Thus far, the intermediary elements of component modeling have not been employed so that the geometry used is only a geometrical surface without any further information. An example of a component (C pillar) is given in Figure 6-3 with the various operations to be performed.

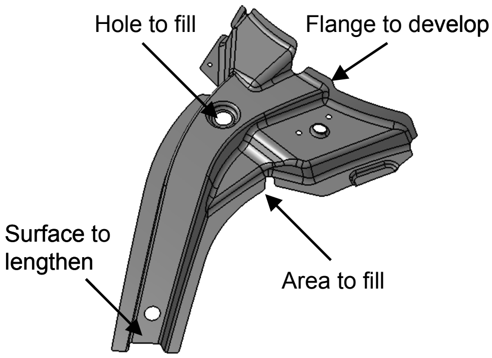


Figure 6-3: Example of a component model.

The corresponding die face is shown in Figure 6-4. The die face model is based on three main domains:

- In the middle, the geometry is quite similar to the component geometry.
- The outside represents the blankholder's geometry.
- Between both previously cited elements, there is a connection surface called addendum.

These three points are elaborated below.

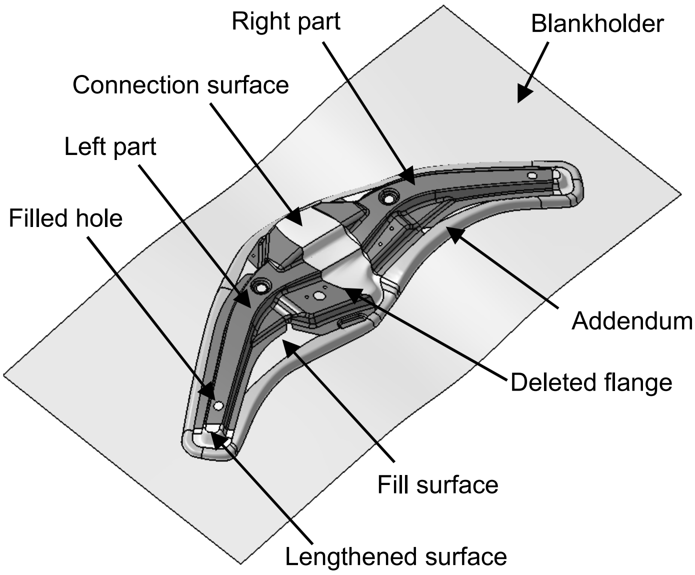


Figure 6-4: Die face model.

Middle domain

In the middle domain, the shapes of the die face follow those of the component except that the holes and the opening are filled and that the shapes are eventually cambered. Holes have to be filled due to the fact that the die face represents the surface of the stamping tool that is in contact with the sheet metal. Thus the component's holes need not be considered in this step. They are treated further for the design of the trimming tool.

The cambering of the component shape is justified by the fact that some sheet metals flatten out under the effect of their weight after the stamping step. To compensate this effect, the curvature of the die face shape has to be increased so that the sheet metal achieves the right definitive shape after its deformation. However this difficulty occurs only in the case of flat parts such as the car roof.

Outer domain
The outer domain represents the blankholder surface: it is the part of the die face where the sheet metal is pinched to draw it on the punch. The geometry of this surface is defined on the basis of designer experience and is validated by the simulation. It depends upon the component geometry but, in the CAD view, there is no direct link between the component and the blankholder surfaces: the blankholder surface is a new element that is generated independently of the component surface.

From a functional point of view, the blankholder shapes slightly follow the component shapes but are smoother. The blankholder may contain an element called a draw bead whose role it is to locally break the material flow in the stamping stage. Its influence is determined in the simulation software as well.

Connection domain
The addendum makes the connection between the component and the blankholder surface. It has an important function in the stamping stage as it determines a large part of the drawing behavior and influences the component quality. The addendum is typically made of a flange going around the component and having a given angle to the stamping direction. The intersections between this flange and the component and blankholder surfaces are rounded.

As explained in section 3.3, the initial definition of the die face is done in the simulation software for the prototype design. The addendum is created on the basis of curves (or profiles) representing the section of the addendum. The addendum surface follows the outline of the component smoothly with some restrictions; yet if the component outline is too complex, it has to be simplified by adding a filling surface. If there are two components to be formed within the same stamping tool, they also have to be connected by means of a filling surface.

The case of flanges located on the outline of the part requires more attention: flanges whose angles relating to the stamping direction are too close cannot be formed in the first stamping operation. Thus the part that comes out of this operation does not contain these flanges. As the component model represents the finished geometry, it has to be modified to correspond to the shape of the die face. Concretely, flanges have to be developed or modified as represented in Figure 6-5.

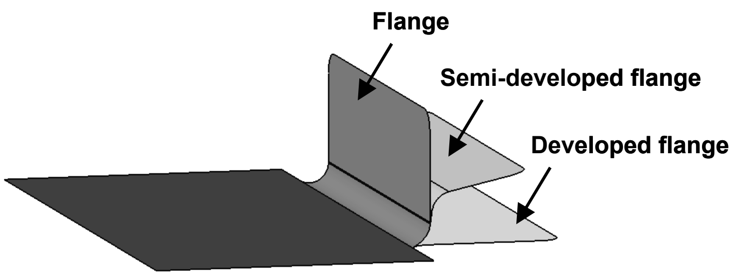


Figure 6-5: Flange development.

In this illustration, a flange is represented in its normal state. Typically, in prototype production, where each edge can be cut with laser independently of its orientation, and in volume production, the flanges have to be fully developed as indicated in the figure. In volume production in some cases, the semi-developed flange may be used when the flange is formed in two steps. In this way, the edge of the flange can be cut before bending it completely.

To model such developed or semi-developed flanges, designers use the component surface as a foundation and extrapolate it. Then they calculate the projected length of the flange graphically or with the help of simulation software to know the outline that is to be cut and to determine the position of the addendum around the component. However, there are difficulties while extrapolating surfaces. In fact, extrapolations may lead to twisted surfaces when the surface to extrapolate and its edge have complex curvatures.

Note 6-1
Even if there is no flange to develop, the component surfaces have to be extrapolated in order to have the necessary surface length for the fillet to be designed with the addendum surface.

6.3.2 Definition of the Engineering Template Behavior

6.3.2.1 Global Situation

Once the properties of both the models of the component and of the die face have been analyzed, the behavior of the engineering templates to be created should be defined. There are still improvement potentials in the component modeling method, above all, in the view of the future die face design. The first observation is that unnecessary activities are carried out and designers do not take advantage of the possibilities offered by CAD software, for example, the fact that the history of the modeling is available in the structure tree. Second, some operations do not yield satisfactory results such as the surface extrapolation.

6.3.2.2 Basic Shapes

Basic shapes are the main shapes of a part. They are common to the component and the die face models. They are the foundation of an engineering template and are specific to each category of component.

6.3.2.3 Form Details

Form details are elements used to further specify the part. They are inserted in the model with the feature methodology and use the basic shape mentioned above as support. As they are present in both the component and the die face model, they have to be introduced in the design earlier than elements specific to the die face. Whether they should be included in both templates depends on whether or not they represent generic forms. If not, designers have to insert them in the product definition stage. Because of their particularity, holes are not included in this category.

6.3.2.4 Holes

Holes are modeled in the component design stage and have to be filled in the die face design stage. This situation could be changed if the modeling sequence of both things would be coordinated. The idea is to make an intermediary state of the component model available for the die design, where no hole is present. That is why holes should be created all at one time at the end of the modeling. This would provide the die face designer with a surface of component without holes, thus avoiding filling them. Holes that have a functional meaning are modeled as features. Other types of opening can be freely created.

6.3.2.5 Flanged Holes

The case of flanged holes is more complex than that of the normal holes since their insertion in a model implies two operations:

- Insertion of a surface (mostly a cylinder)
- Trimming of the support surface

Hence, if we extract the intermediary state of the model that does not contain the flanged holes, the surface is not trimmed, but the shape of the flanged hole will be missing.

In a traditional modeling method, for the die face design, flanged holes are closed by a rounded surface, thus flanged-hole have two different views depending upon the context. That is why they are modeled as multiple-view features as described in section 5.1.3. Such a feature is presented in Figure 6-6. In the component view, it is a normal flanged hole but, in the die face view, the geometry is completed by a closing surface representing the intermediary stage of the sheet metal after the stamping operation and before the hole trimming. Both items are logically linked so that the closing surface geometry is dependent on the flanged hole geometry in the case of modification.

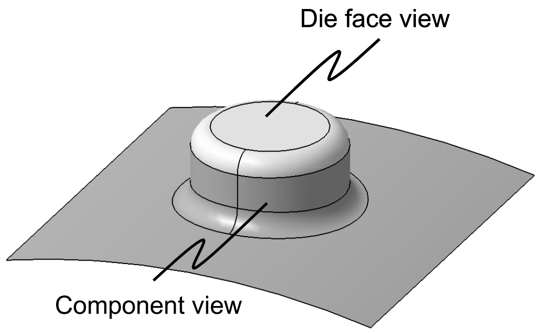


Figure 6-6: Flanged-hole multiple-view feature.

6.3.2.6 Surface Extrapolation

In the same way as for the holes, lengthening of the component surface could be avoided if die face designers used an intermediary state of the component model where the surfaces are not already cut. Also, the complete outline would be made in one single operation instead of cutting the surfaces step by step at various times in the modeling sequence. This is illustrated in Figure 6-7 and Figure 6-8 where two generic scenarios are given: Let us assume that the surface to be created is made of three basic surfaces. The final outline is made by cutting the surfaces. In the first figure, each surface is cut and the results are combined. As a consequence, there is no intermediary state of the design where all the shapes of the three surfaces are present without having been cut. The second solution is therefore more advantageous: in the second represented sequence, the surfaces are first assembled and are cut afterwards so that the uncut surface can be used for the die face design.

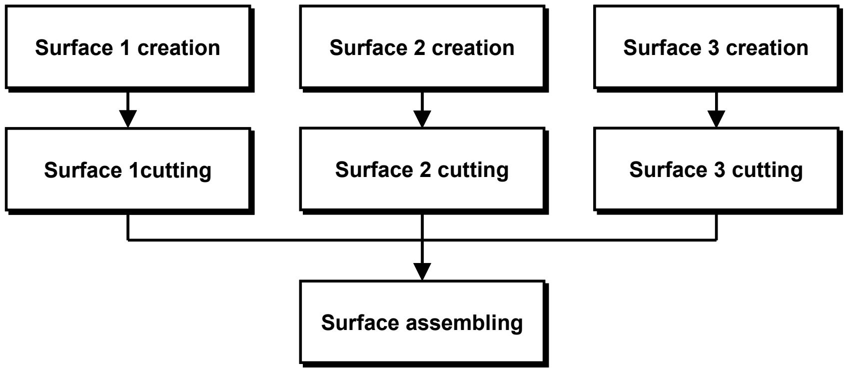


Figure 6-7 Parallel surface cutting

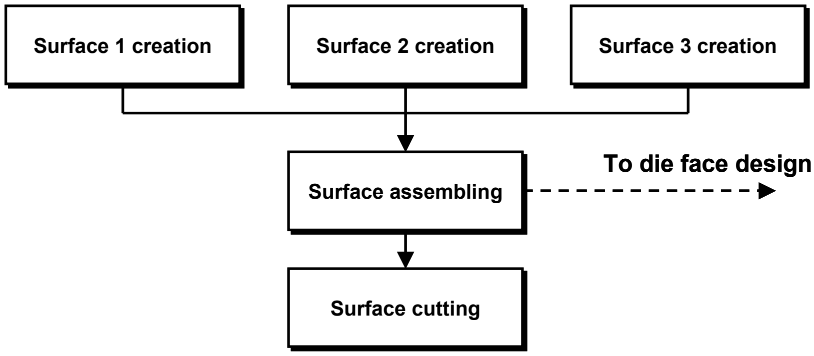


Figure 6-8: Single surface cutting.

Note 6-2
To simplify the representation, only the elements necessary for the demonstration have been represented.

6.3.2.7 Flanges

As mentioned above, there are three kinds of flanges:

- Flanges to be kept: flanges present in the die face
- Semi-developed flanges: flanges present in the die face but in a different form
- Developed flanges: flanges not present in the die face

The fact that flanges are present or not in the die face determines the order they have to be modeled. For the same reason as earlier, die face designers should have access to a version of the model where only the flanges they wish to keep are available. Thus to avoid having to delete undesirable flanges, these should be made after those that have to be kept. This procedure also avoids extrapolating the surface at the place of the deleted flanges, as it is already uncut.

In the case of a semi-developed flange, there is not an intermediary version of the component representing its view in the die face since semi-developed flanges have another shape. Thus this version should be created explicitly. This issue can hence be supported by multiple-view features, which may have views of both the normal flange and the semi-developed flange.

In conclusion, a first modeling sequence can be identified for the component model: basic surfaces → shape detail → flanged-holes → flanges → outlines → holes.

Note 6-3
Since models are made with the engineering template methodology, the modeling sequence is visible as the time order in which the elements have been created but rather as logical relations between them.

The sequence given here has been set up to optimize the derivation of the die face model from the component model; yet there are other requirements for the component template that are justified by the optimization of the component design itself. Above all, it is necessary to define how accurate the component template should be, what the designer's possibilities are, etc. In the following, we will focus our attention on the part of the component template that influences the die face template.

Table 6-1 summarizes the function of each type of the elements cited above. They are regrouped by categories as follows:

- Common geometry: gathering elements present in both templates of the component and the die face in the same form
- Hybrid geometry: gathering elements present in both the templates of the component and the die face but in different form
- Component geometry: gathering elements being exclusively present in the component template
- Die face geometry: gathering elements being exclusively present in the die face geometry

Function	Elements
Common geometry	Basis shapes, form details
Hybrid geometry	Flanges, flanged-holes
Component geometry	Holes, contours
Die face geometry	Blankholder, addendum

Table 6-1: Classification of the component and die face elements per function.

6.3.3 Engineering Template Modeling

6.3.3.1 Introduction

In the next section, we will touch on the topic of engineering template creation and use. Thus words such as "creation" or "definition" refer to the engineering template development and "parameter setting", "template modification", "change", and "adapting" refer to the use of the template by designers.

6.3.3.2 Model Associativity

Once the contents and of the templates and the relationships between them have been defined, the problem is to model them concretely. To link the templates of a component and its corresponding die face, two solutions are used:

- Reuse of intermediary states of the component, as developed above
- Linking of die face's exclusive elements with the component's elements

The linking of die face elements and component elements is done using rules in the CAD system. In particular, the following elements have to be linked:

- The addendum and component's outline
- Blankholder shapes and component shapes

6.3.3.3 Control Curves

As surfaces are generated with help of curves, curves are directly used to implement the rules. They represent the skeleton of the component and die face models.

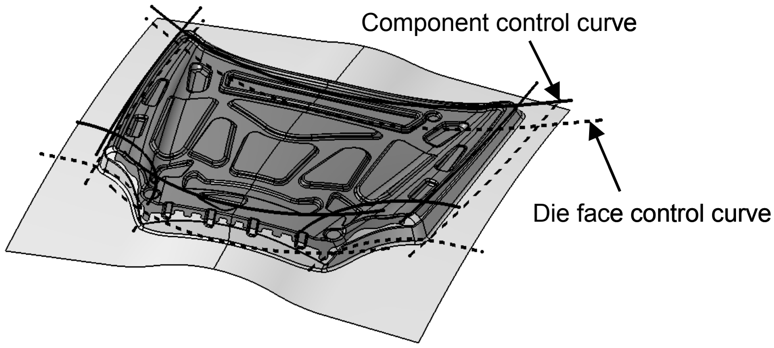


Figure 6-9: Addendum control curves.

In Figure 6-9 control curves of the component outline are depicted as well as the derived addendum control curves. A modification of the component curves leads to a modification of the die face curves and thus of the die face surfaces. If the template behavior is not satisfactory for the designer, the die face curve should not be directly modified: the rules should be adapted.

For example, the die face control curves on the rear are parallels to their respective component control curves. Consequently, there are two alternatives for modification of the addendum: change the component curves or change the parallelism rule (amend the value of the parallelism or replace it by another type of constraint).

The blankholder surface is linked to the component surface in the same way as for the addendum. An example for one curve of the blankholder is given in Figure 6-10. In this case, the blankholder curve relates to two curves of the component. The rules are expressed in the form of distances between the curves.

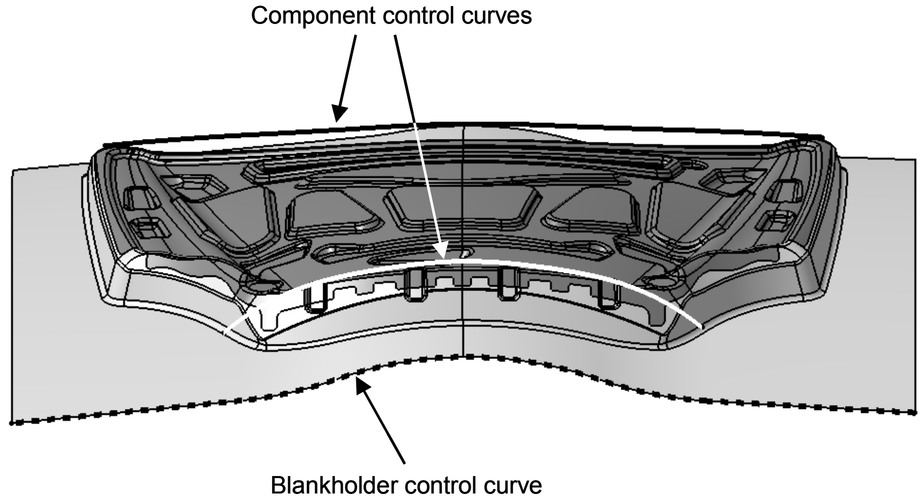


Figure 6-10: Blankholder control curves.

6.3.3.4 Surfaces

With the analysis of the component and die face relationships, it appeared that the die face is based on uncut component surfaces. That is why we can consider the component surface as being a cut-out of the die face rather than considering the die face as a lengthened component surface. It is hence better to derive the component surface from the die face surface rather than vice versa. This may seem illogical in the actual design process but this is made possible by the use of templates.

Consequently, there are bidirectional influences between the component model and the die face model:

- Component control curves lead to die face control curves.
- Die face surface leads to component surface.

This results in the relation loop depicted in Figure 6-11, where it is evident that component surfaces are derived from die face surfaces. For the component design, the result is exactly the same as if this surface were derived directly on the basis of the component's control curve; but this way of working enhances the quality of the die face by avoiding surface extrapolation and establishes an associative link between both templates.

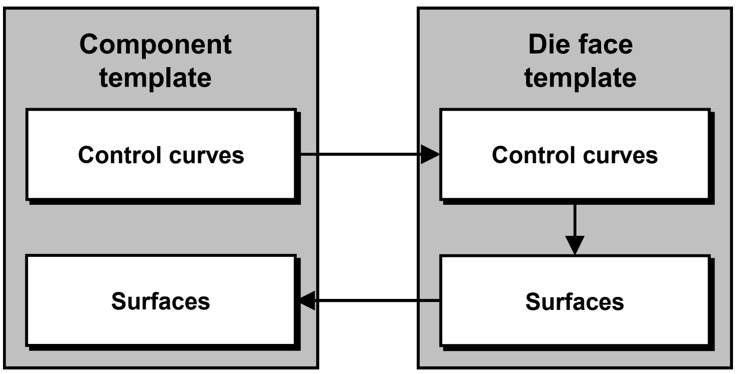


Figure 6-11: Control curves and surface relations.

6.4 Integration of Simulation Information

6.4.1 Simulation Information for the Component Template

In the stage of the information acquisition, the geometry of the die face has been validated by the simulation (Figure 6-12). The aim is then to profit from the knowledge resulting from this operation to improve the behavior of the templates and thus to improve the quality of the products.

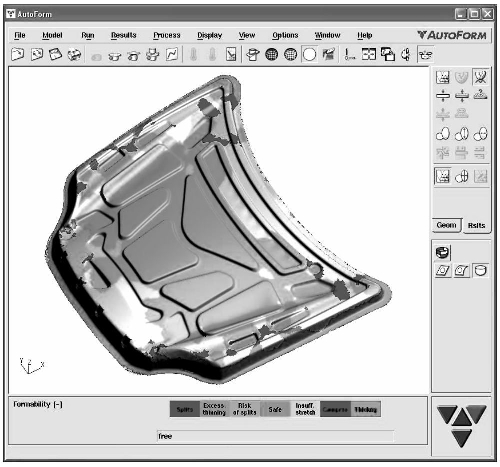


Figure 6-12: Result of the simulation of the hood in AutoForm.

The first information from the forming simulation that should be integrated in the template is the stamping direction. Indeed the orientation of the geometrical elements in the component is necessary to assure its formability. In the traditional design process, the punch direction is initially estimated by the designer and then it is validated in the simulation stage. But the simulation takes place once the model is achieved or greatly advanced so that many corrections have to be made if the punch direction is not correct. If a validated punch direction is introduced in the component template, it can be directly used as input for the modeling, and errors are avoided. Thus the design is no longer validated by the simulation but is rather driven by it.

However, designers have the possibility to modify the template geometry or the punch direction, making it important to define the limits of the modification field. Several tests hence have to be performed by modifying the design parameters and defining the range of values that lead to a result validated by the simulation. In this way the user of the template could be informed when a configuration leads to a non-validated result.

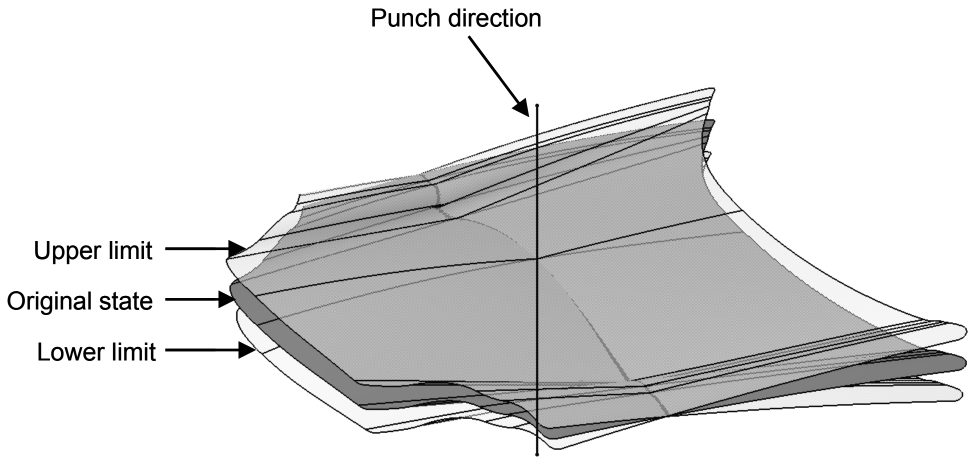


Figure 6-13: Parameter range of validity.

Figure 6-13 illustrates the range of the user's possibilities for the determination of the hood's basic shape in relation to the punch direction. Further details of the model are adapted to the punch direction in the same way.

Another solution would have been to adapt the punch direction to the geometry instead of adapting the geometry to the punch direction. Concretely, the orientation of the punch direction would be defined in relation to the orientation of the component geometry, for example, of the component's basic shape. Nevertheless a modification of the punch direction implies renewed modification of the rest of the geometry: the orientation of the details would have to be adapted, leading to possible conflicts with other design requirements.

6.4.2 Simulation Information for the Die Face Template

Simulation information should be integrated in the die face template as well. First of all, the simulated punch direction is useful for the definition of the overall die face. The addendum and the blankholder geometries have been defined and validated in the simulation system. Thus the geometry of the template in its original configuration is already validated by the simulation. The question is then whether it is possible to predict the validity of the geometry when parameter values are changed by designers, as demonstrated for the component geometry. Yet two issues make reaching this goal difficult. First, it is not possible to envisage all the possible configurations for the die face as there are too many. This is due to the fact that the addendum depends not only on the component geometry but also on the blankholder geometry and on its own parameters. Second, the addendum should be defined precisely and cannot be done without the validation of the simulation. Moreover, as section 3.3.2 explains, the simulation software enables the users to generate and modify the die face geometry quickly.

For all these reasons it is more interesting to develop an efficient coupling between the CAD and simulation applications rather than trying to insert too much simulation information in the CAD model. The next part is devoted to the description of the CAD und simulation models coupling.

6.4.3 Coupling of CAD and Simulation

Now (with the engineering template methodology), the associativity between the design and the simulation resembles that shown in Figure 6-14, which sets out the associations between the component and die face templates and the simulation model.

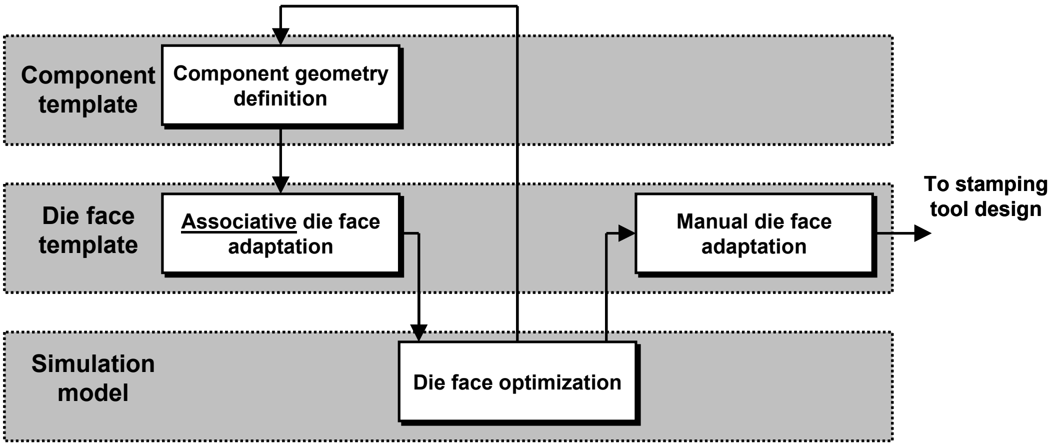


Figure 6-14: Coupling of CAD and simulation models with manual adaptation.

A modification of the component template automatically involves a modification of the die face template. But it deals with the adaptation of only a part of the elements of the die face, as not all of them can be derived from the component. In fact, the component geometry determines the outline of the addendum and partially the shape of the blankholder but not the shape of the addendum, which is a question of the stamping technique. As a result, a part of the die face definition has to be done in the simulation system and the die face CAD model subsequently has to be manually adapted by designers.

A cooperation project has been carried out with the company AutoForm Engineering GmbH, which develops a commercial application for the design and the simulation of stamping tool. The aim was to define the requirements of the automotive company DaimlerChrysler AG for the simulation software.

The project came to following result: the associativity is obtained by using the same model in both the CAD and simulation software. This implies that the simulation system must be able to read the CAD model and that the geometry in both systems must be modeled with the same methodology. In this way it is possible to modify the die face model in both systems and to simulate the result without a break in the process chain. Second, thanks to the associativity between the component and the die face models, modifications of the component geometry can be immediately validated by the simulation.

As the modeling of the die face in the simulation software is based on profiles, this methodology had to be adopted for the die face CAD model. First, parameterized profiles have been developed and should be identifiable and settable by the simulation systems. Then a first modeling methodology has been envisaged and consists of generating swept surfaces directly with these profiles. But this proceeding does not yield the same shape as that generated by the simulation software. In particular, circular edge transitions do not keep a constant radius value, as they should (see Figure 6-15).

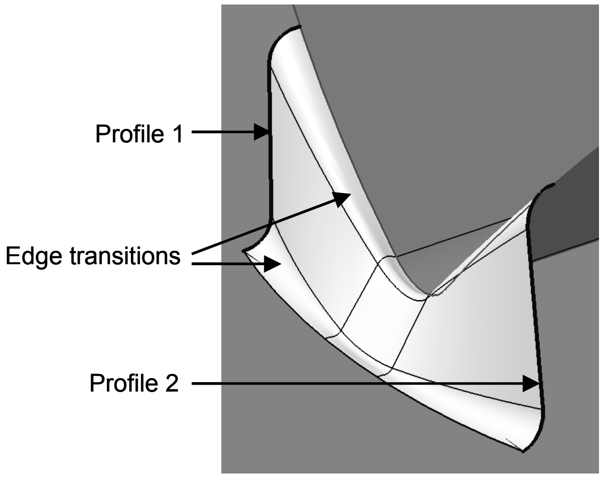


Figure 6-15: Profile based addendum modeling.

Consequently, profiles are used to generate the addendum surface but not directly: only the profile parameters are employed as reference for the modeling of the addendum which is based on two surfaces and two fillets as shown in Figure 6-16.

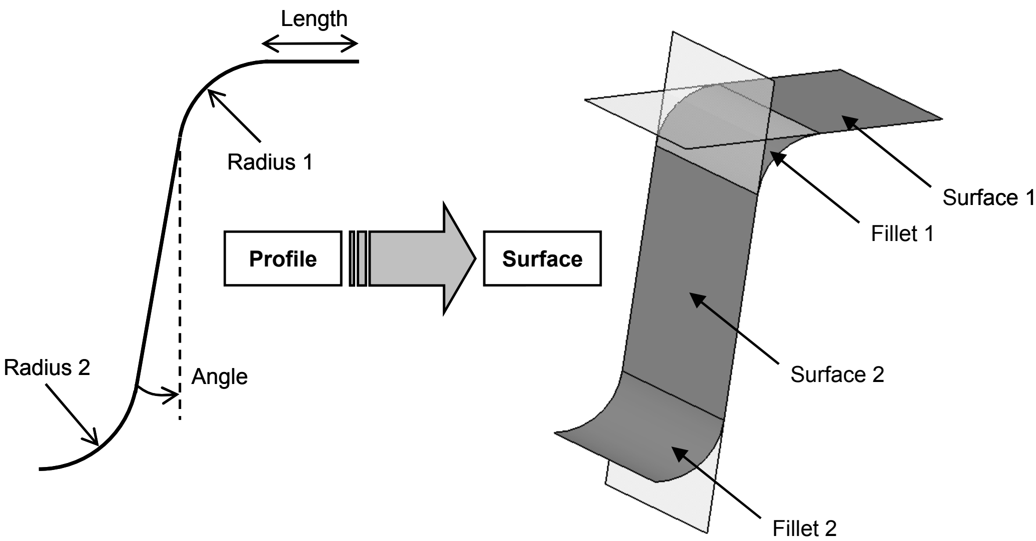


Figure 6-16: Surface based addendum modeling.

The parameterized features and the modeling methodology have already been developed and validated for this thesis but the implementation of the described solution remains to be carried out by the simulation software provider. Thus, in the future, the coupling between the die face design and the simulation step should work as depicted in Figure 6-17.

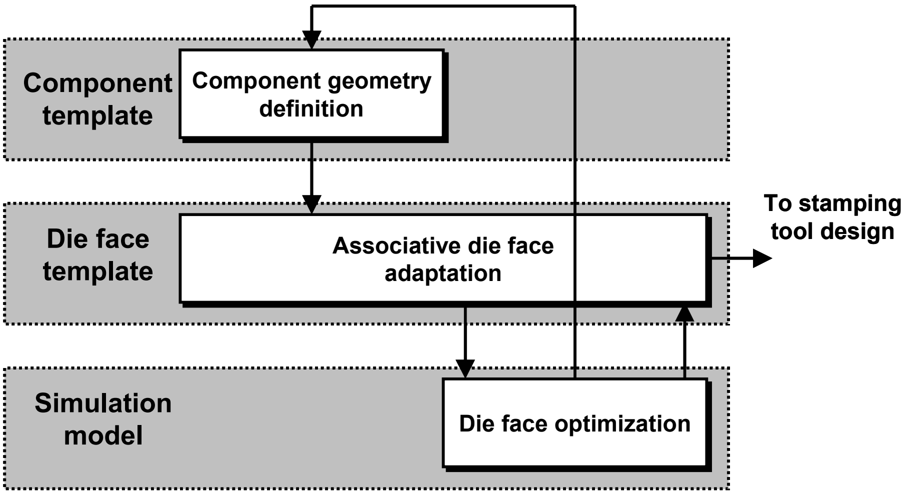


Figure 6-17: Coupling of CAD and simulation models with associative adaptation.

In the same way as described above, the associativity between the component geometry and a part of the die face is already guaranteed. The improvement should lie in the fact that further elements of the die face being tuned in the simulation software are also directly adapted in the CAD model.

6.5 Result of the Engineering Template-based Hood Model

The modeling method of the hood's die face is displayed in Figure 6-18. There are three parts: the common geometry at the top and the derived models of the component and the die face. The common geometry contains the basis of the component and the die face, plus the intern form details. The component's outline, which is also the profile curve of the addendum, belongs to the category of the common geometry as well.

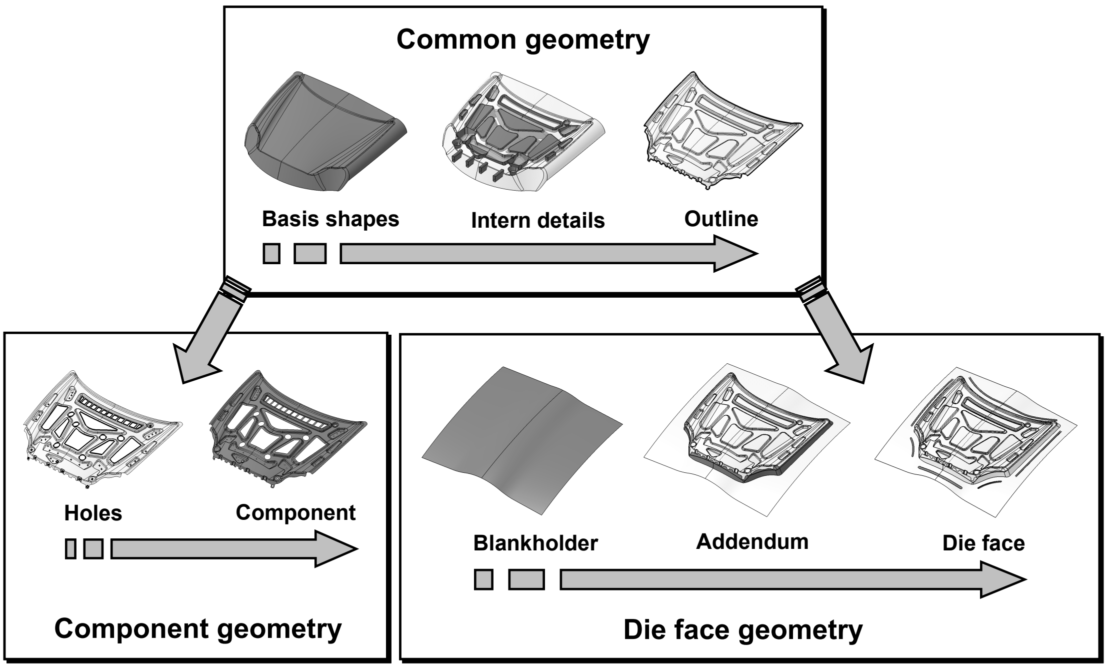


Figure 6-18: Hood component and die face templates.

On the left-hand side of the figure, the finishing of the component model is shown. In the case of the hood, only the holes are missing. They are all inserted at one time as curves, and the surface is then trimmed to accomplish the finished component model. On the right-hand side, the picture shows the modeling of the die face based on the blankholder, the addendum, and the draw beads to achieve the design.

Chapter 7
Stamping Tool Engineering Template

7.1 Introduction

Stamping tool design represents the link between car-body component design and volume production. Because of this, the stamping tool design step has many inputs to consider but also many requirements to meet from the production perspective.

This chapter aims at describing the engineering template methodology applied to the stamping tool design process. Whereas chapter 6 described a template based on surfaces and the integration of simulation information in the design, this methodology focuses on solid modeling and the integration of standards and process information.

7.2 Requirements

7.2.1 Introduction

The requirements imposed on the template development are justified not only by functional problems but also by the systems used to support them and the targeted users, as depicted in Figure 7-1. The following sections develop these issues.

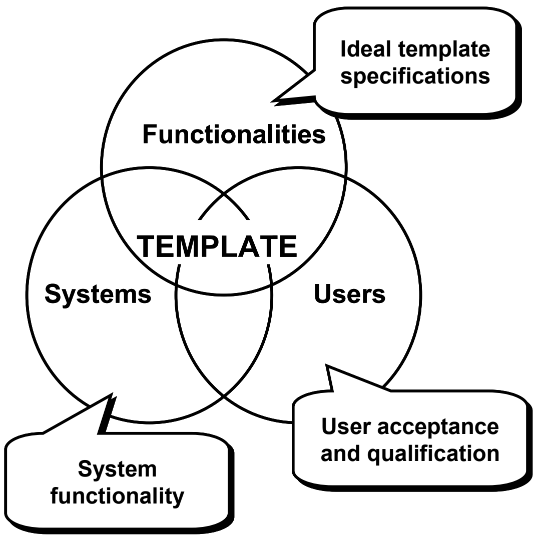


Figure 7-1: Engineering template requirements.

7.2.2 Functional Requirements

As mentioned in section 3.4.3, stamping tool design obtains its inputs from component design, while also requiring information from the die shop, which provides the description of the presses. In fact, designers have to develop a stamping tool that matches the die face and the dimensions of the press to be used for the production.

The choice of the fitting press for a given component is made in the production facilities planning and does not depend upon the component but rather on the press line capacity and availability. That is why it is not possible to assign a component to a fixed press or to predict it.

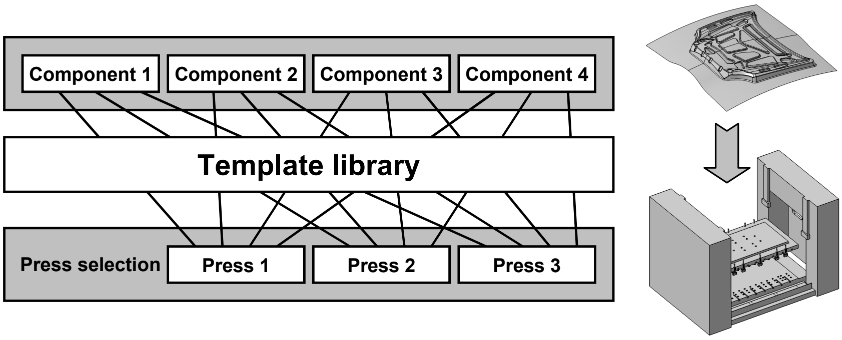


Figure 7-2: Die face and press combinations.

The challenge for template development is to build a template library covering all the possible design configurations (Figure 7-2). For example, 15 press lines could be employed for 116 components, which would theoretically result in 1740 possibilities if there were no similarities between the presses.

The ideal case would be that a single template covers all the possibilities. Nevertheless we have seen in section 5.2.2 that the more functionality templates have, the more they are complex for users and the more they require hardware performance. But simplifying the template's behavior also means increasing their number. An alternative would consist of creating one or a few templates that do not define the overall product geometry and that will have to be completed by the designers.

7.2.3 User Bias

The experience gained with designers who were to use templates has shown that they prefer working with templates that do not fully achieve the design but are totally reliable to working with templates describing the complete product geometry but requiring more adaptation and parameter setting. This is due to the fact that a complex template would be delivered at one time and would have many parameter values to adjust so that the relationships between them and the order they have to be changed would not be easy to grasp.

This situation leads to the idea to develop a modeling method that allows a product model to be built step by step so that designers retain the impression of having created the geometry and would have the possibility to adjust the parameter values one by one. In this way, designers gain a better understanding of the model structure and parameterization.

A second argument that strengthens this notion is that designers often do not accept a predefined geometry if it is not fully correct. Two examples illustrate this:

- Templates describe almost the complete product geometry but require corrections to adapt them to the current project
- Templates describe only half the product geometry but are fully correct. Then designers will have to specify the remainder of the geometry.

If we compare both alternatives, it appears that the second one requires more time than the first one since, instead of adjusting parameter values with the integrated knowledge supporting designers in their task, designers would have to add new geometrical elements to the model. Still, designer bias is lower in this case due to the fact that no correction is required and the understanding of a complex template requires a greater expense of time.

7.3 Modular Design

As a consequence of the arguments listed in the last sections, two decisions have to be taken:

- Definition of the template size: should the entire stamping tool be made of one template or should it be the sum of several templates?
- Definition of the template detailing level: how far should templates be detailed?

To fulfill the designers' expectations, both questions have to be answered. Table 7-1 summarizes the possible combinations.

Configuration \ Detailing level	High	Low
Single template	1	2
Several templates	3	4

Table 7-1: Level of detail of the engineering template.

Combination 1 corresponds to a fully detailed single template. For the reasons already cited, this solution is not the best one from the view of the user, although it is interesting for the template developer.

Combination 2 corresponds to the case where a single low detailed template is employed for the overall product geometry. This solution suits the requirements of the users that do not wish to have to correct the geometry delivered by the template, but the model still has to be further detailed.

Combination 3 corresponds to the modeling of a product based on several templates each having a high definition level. In this way, designers would have to choose fitting templates in a library and to assemble them. The advantage of this solution is that designers keep the control of their models, but the application of such a method is difficult because of the too great variability.

Combination 4 is the case where several templates having a low definition level would be used. This solution does not offer any advantage in comparison to the others as it is neither quick and efficient, nor does it deliver a result of good quality in the sense of knowledge integration.

To sum up, the second combination consisting of using one template for the whole product definition but with the necessity to detail the model is the best one for utilization in practice. The question is then how the detailing phase should be carried out. In section 5.5 the combination of templates, integrated components, and features was described. This is the solution that will be applied for the detailing of the model since details missing in the templates are often standard elements or parts. Table 7-2 shows the method associated to each element type for the chosen solution.

Element type	Associated method
Basic shape	Template
Standard part	Integrated component
Detail	Feature

Table 7-2: Element type and associated methods.

7.4 Stamping Tool Engineering Template Definition

7.4.1 Input Data

As mentioned above, an engineering template should be adapted to its inputs represented by the die face and the press. These elements are a part of the skeleton necessary for the engineering template use. Together, they define the main template dimensions: the dimensions of the die face define the minimal stamping tool length and width values, whereas the press table dimensions define the maximal values. In fact, as material has to be saved, the cited dimensions of the template will be directly linked to the die face to make the stamping tool as small as possible and the dimensions of the press will be used as limit values.

The press characteristics also define the maximal stamping tool height, and the flow chart defines the height of the die face position in the stamping tool. Further information is taken from the press geometry such as the position of the clamping slots or of the centering holes. To summarize, the die face defines the core of the stamping tool and the press defines the outside.

Designers used to insert the press geometry in their model for the visualization. This geometry is completed by reference elements, which are the necessary inputs for the template. As a consequence, each press of the die shop has to be modeled in this way. This is an additional task required by the implementation of the template methodology, but it has to be performed only once. Moreover the press lifetime is several decades and there are often similarities between presses so that it is also possible to create a single parameterized model for several presses where the parameters are taken from the standards.

The insertion of the die face and the press information constitutes the first part of the skeleton definition. The rest of the skeleton is derived from this information, with reference to the standards and second to the user's knowledge. The skeleton regroups the sub-skeleton for the template itself but also reference elements for the rest of the geometry that is to be modeled, as mentioned, using integrated components and features. This principle is illustrated in Figure 7-3, where the three skeleton levels are shown with their relationships. The rules and relations between them are depicted with the symbol f(x), and the geometry positioning relating to the skeleton is represented with a target symbol. It appears that the detailing skeleton depends upon both its master skeleton (template skeleton) and the template geometry, as modification in the template may affect the positioning of the details.

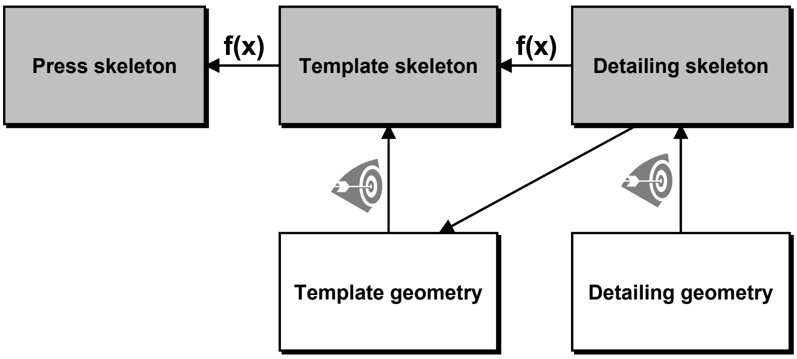


Figure 7-3: Skeleton and template relation for the stamping tool model.

7.4.2 Parameterization

The parameterization is necessary to adapt templates to the current design but also to enable designers to modify it.

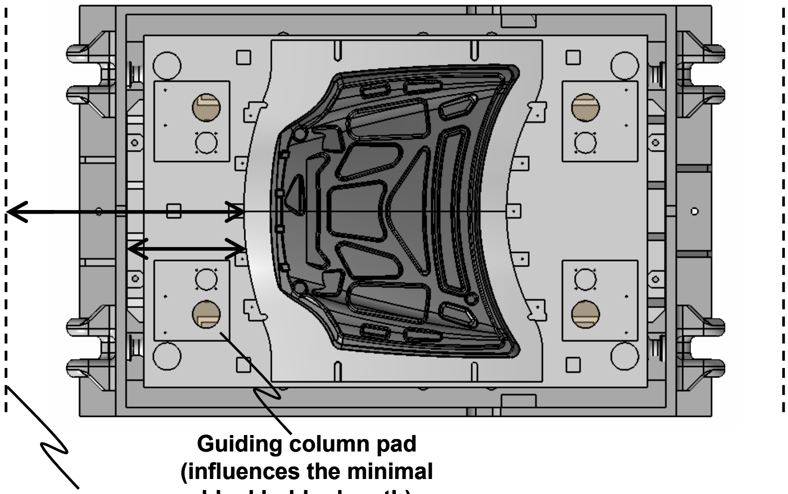


Figure 7-4: Stamping tool template parameterization.

During the template insertion or modification, parameter values are set depending upon the context and upon the related rules. But in many cases there still be the possibility for users to change them under some restrictions. The restrictions are imposed not only by the standards but also by the context; that means that the range of values for a parameter is defined dynamically. As illustration, Figure 7-4 represents the top view of the assembly composed of the punch, the blankholder, and the lower die. The template automatically sets the distance between the die face and the edge of the lower die to the lower possible value. This depends upon the dimensions of the guiding columns to be inserted in the blankholder. Yet the upper value is contingent upon the situation: it is the difference between the lower die length and the die face length. Thus the maximal value is reached when the lower die length is the greatest and die face length is the smallest. In addition, the upper value for the lower die length depends upon the press dimensions, with the lower value for the die face length depending upon the component dimensions. In this case, the lower die length is automatically set by the template to the minimal value; the user may change it afterwards within the mentioned range, if desired.

7.4.3 Result of the Stamping Tool Engineering Template

Figure 7-5 shows the result of the stamping tool template applied to the example of the hood. In the same way as described in the previous section, the overall geometry has been parameterized on the basis of the information associated with the die face and the press. There are, in all, five main components whose basic body is described. In addition, elements that can be fully defined in this stage are present in the template as well: ribs, blankholder feet, and lift lugs.

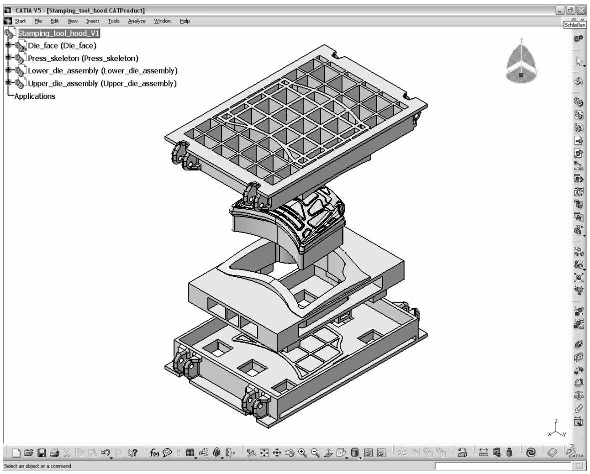


Figure 7-5: Stamping tool template applied to the hood model.

7.5 Process Information

7.5.1 Standards

This section sets out how to integrate information about standards into a model. The better solution is to act on the dimensions of the geometry. For this, the skeleton has to incorporate standard information: as the geometry is based on the skeleton, controlling the skeleton is equivalent to controlling the geometry. Thus the relationships between reference elements have to contain restrictions about their modification possibilities. Three categories of restrictions are given below, together with a corresponding example:

- Locked value: It is the case where users are not allowed to modify the parameter value. The distance between both planes defining the thickness of the lower die table amounts to 60 mm. As a consequence, users should not have the possibility to manipulate this value, but if one of the planes is moved, the second one will automatically be moved.
- Free value within a given range: For some elements, designers are free to choose a value but within a given range. That means that the lower or higher possible value (or both) is imposed. In this case, designers can enter any desired value as long as they do not cross the limit. If they do, a message informs them of the maximal or minimal possible value. For example, designers are free to define the stamping tool height up to a given value represented by the press opening.
- Limited number of values: Holes have a limited number of diameter values corresponding to standardized screw and drilling tool dimensions. Designers have to select the desired standardized value from a list. Holes features have to contain these values and display them in a scrolling list, for example.

7.5.2 Multiple-View Model

A CAD model has to be delivered to different actors who are not interested in the same information. That is why the model should contain different views, in the same way as for the car-body component model, which contains information for the die face design.

The template structure foresees the subdivision of the model into various functions. In particular, several views of a component are available, corresponding to different applications. For components that are to be cast, three views are required:

- The completed component model is needed for the visualization of the future product and the examination of the assembly working as well as for the representation of the geometry for the finished component control.
- The rough component geometry corresponding to the geometry of the part to be cast is needed to produce the polystyrene model used as core for the casting.
- The geometry of the machined volume is necessary for manufacturing engineers to identify the operations to be performed on the basis of the CAD model. That is why the information needed is regrouped and structured in an accessible way. Volumes to be removed by cutting are modeled separately from the rest of the geometry and represent the difference between the rough part and the finished part. In the same way, holes are modeled in separate containers and are sorted by dimensions and machining direction, so that the first step of the process planning is prepared.

Some of the information mentioned is not present in the base template and is inserted afterwards, for instance, the holes. In this case, the place for these elements in the structure is prepared but is left empty. It is filled in by the designer during the specification phase. All the views are linked to the same parameter set belonging to the definition of the finished component view, as it is the one that designers are interested in. Thus, the designer only considers the finished model and the other views are adapted associatively.

Note 7-1
The presented methodology is valid only for the cast components. For others, which are often standard components, there is no need to define multiple views.

7.5.3 Machining Information

Information for the machining quality may already be inserted in CAD models. Templates integrate this kind of information as well when the related geometry is already defined. In other words, tolerances are also inserted in templates without requiring additional work from designers. Depending on the desired representation solution, tolerances can be displayed by coloring the relevant surfaces or by inserting alphanumerical information.

As mentioned in the second part of the thesis, engineering templates do not suppress the use of features. That is, methods consisting of using features as information carriers for the machining planning are still valid and that is the reason why this work does not elaborate on the insertion of machining information into CAD models.

7.5.4 Documentation

After the modeling of the stamping tool, a set of documents is to be created. With the introduction of engineering templates, the procedures are modified as explained in the following. First, the bill of materials can be generated directly with the software features and the standardized model structure facilitates the work. Second, the completion of the check lists can be partially automated thank to the use of semantic elements and of the template, which has some built-in control mechanisms to avoid errors. More details about this topic are given in section 7.6.3.2.

7.6 Detailing of the Design

7.6.1 Introduction

The template described in the foregoing section does not describe the entire stamping tool geometry. As mentioned, this geometry should be complete by integrated components and functional features. This section illustrates the use of components and features for stamping tool design.

7.6.2 Integrated Components

7.6.2.1 Component Contents and Geometry

Among the elements missing in the stamping tool model, all the ones that are composed of a sub-assembly of components are modeled by the methodology of integrated components developed in section 5.3. For the demonstration of the integrated component definition and employment, we will concentrate on one example, namely the guiding column.

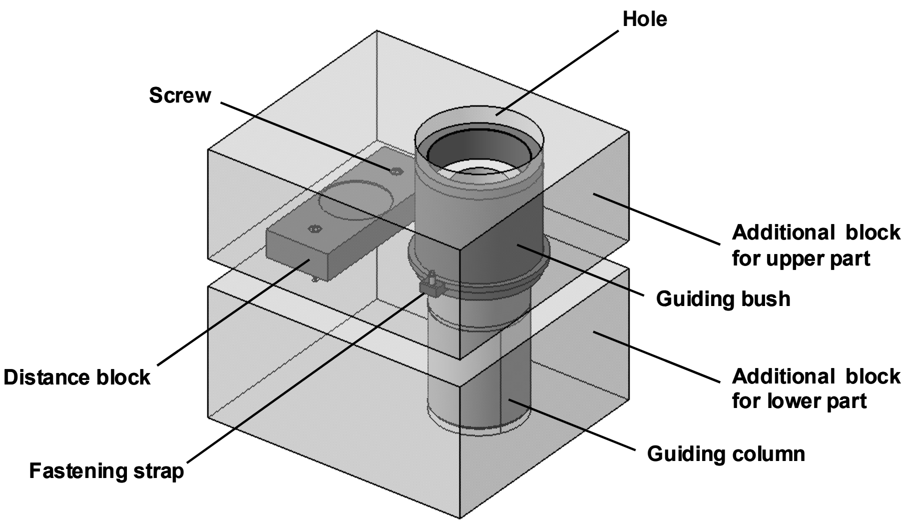


Figure 7-6: Guiding column-integrated component.

As shown in Figure 7-6 the assembly of the guiding column is composed of following elements:

- Main component

    - The guiding column itself
- Additional components
    - A guiding bush
    - Fastening straps
    - A distance block
    - Screws
- Additional elements
    - An additional block for the upper part
    - An additional block for the lower part
    - Holes for the column and the screws

The main and the additional components are added in the structure tree in the assembly level. But the additional blocks and holes have to be distributed in the structures of different components: the lower and the higher part of the die. The difficulty is then to define the right place of these elements in the destination part and to manage the interactions with other geometrical elements. Indeed after their insertion in the model, they are identical to other features. If the model does not undergo any modification after the component insertion, the function of the additional blocks and holes is guaranteed as their geometry is fully preserved: positive added volumes fill the corresponding domains of the model, no matter if these volumes were empty or full and, in the same manner, holes remove material in the just created volumes. Yet, in most of the cases, the models may be modified after the component insertion, either by changing existing features or by adding new ones.

Modification of existing features does not cause problems since the order is kept and thus the additional blocks and holes placed at the end of the relation chain are not modified. But if new features are added, these may interfere with the elements of the integrated component. For example, holes may be partially or fully filled, or a part of the added blocks could be removed, diminishing their function.

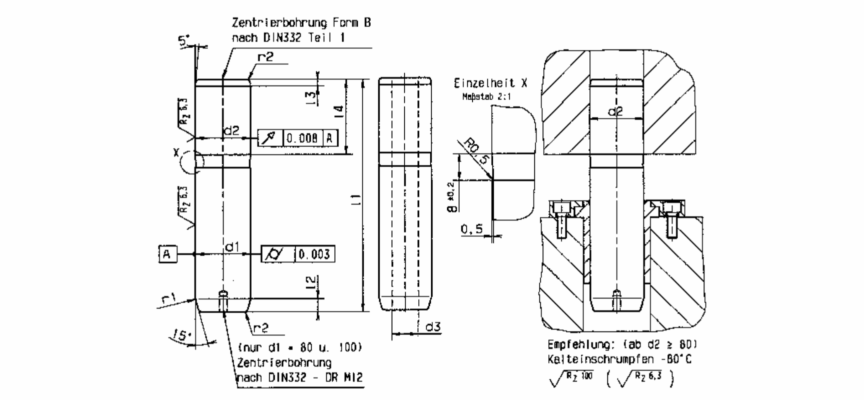
To deal with this problem, a solution had to be found to avoid the insertion of features in an undesirable position: the positions of the geometrical elements within the model are managed by the skeleton. Thus, to avoid geometrical elements overlapping, related reference elements have to be maintained at a given distance by considering the elements' dimensions. However this solution is not failsafe as it does not prevent users from modeling features without making use of the skeleton. The section 8.4.2.2 depicts a possible solution for this problem.

The guiding column's assembly is positioned using an input point that belongs to the skeleton model and is thus dynamically associated to the rest of the geometry. The column axis is placed on the positioning point. The vertical axis of a stamping tool is represented by the normal axis of the model, and a guiding column has to be vertically oriented. Thus, during the insertion of this component, the orientation is set directly. The user is then asked to define the remaining orientation along the symmetry axis of the column.

7.6.2.2 Component Parameterization

As the guiding column is the main part of the integrated component, the dimensions of the adjacent items are defined relating to its dimensions. Thus one master parameter is chosen (the column diameter) and the other ones are automatically adapted. The parameter values are taken from the standards (see Figure 7-7) and are sorted in a design table that serves as reference for the dimension modifications.

The dimensions of the column influence not only the dimensions of the other elements but also their position. For example, the fastening clamps have to be moved when the column diameter increases.



d1 g6	d2 H6/r6	l1 0 -1														d3 +2 -2	l2 +0,8 0	l3 +0,8 0	l4 +0,8 0	r1	r2
		125	140	160	180	200	224	250	280	315	355	400	450	500	560						
25		x	x	x	x	x	x										8	4	40	3	2
32			x	x	x	x	x	x									8	4	45	3	2
40			x	x	x	x	x	x	x								8	4	56	3	2
50				x	x	x	x	x	x	x	x						10	4	70	5	2,5
63					x	x	x	x	x	x	x	x					10	4	80	6	2,5
80						x	x	x	x	x	x	x				40	10	4	100	8	3
100							x	x	x	x	x	x	x			50	10	4	125	10	3
125									x	x	x	x	x	x		65	12	5	140	12	4
160											x	x	x	x		95	12	5	180	18	4

Figure 7-7: Extract from DIN 9833: guiding column dimensions.

7.6.3 Functional Features

7.6.3.1 Features Belonging to the Template

Lift lugs (see Figure 7-8) are elements that can be put into the category of functional features as they do not represent basic shapes of the model but rather detailing elements. However, in section 5.2.2, we have differentiated functional features belonging to the engineering template and others added afterwards. In almost all the cases there are four lift lugs in the lower die and in the matrix. That is why they should be directly integrated into the template. In the rare cases where more lift lugs are needed, they can be added to the model later by designers.

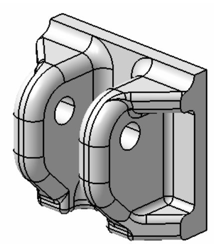


Figure 7-8: Lift lug.

The lift lug dimensions are standardized for reasons of stiffness and safety . The decisive information for the dimensioning is the weight of the stamping tool. The overall lift lug geometry is defined as a function of the weight of the stamping tool. Thus, designers do not have to deal with dimensions of geometrical elements but instead take advantage of a feature whose behavior is adapted to its utilization. In CAD systems, the weight of the modeled product can be calculated. The feature geometry is directly linked to a parameter giving the stamping tool weight, so that it adapts to the modification of the model: if the product weight increases, the dimensions of the lift lugs increase proportionally without designer intervention.

7.6.3.2 Functional Features for Model Detailing

Functional features for the model detailing are the ones that are not directly integrated in the original template. The example of the clamping slot shown in Figure 7-11 has been chosen to illustrate the use of functional features for the stamping tool detail design. This feature contains engineering know-how as well as process and standards information.

Positioning and orientation

Clamping slots are located on the contour of the lower die and the matrix. There are used to fix the stamping tool to the press table.

Clamping slots have not been built into the stamping tool template as their number may vary. Additionally, there are many solutions for their positioning depending upon the press geometry. In fact, the press table contains many grooves from which the designer can choose the desired ones (see Figure 7-9).

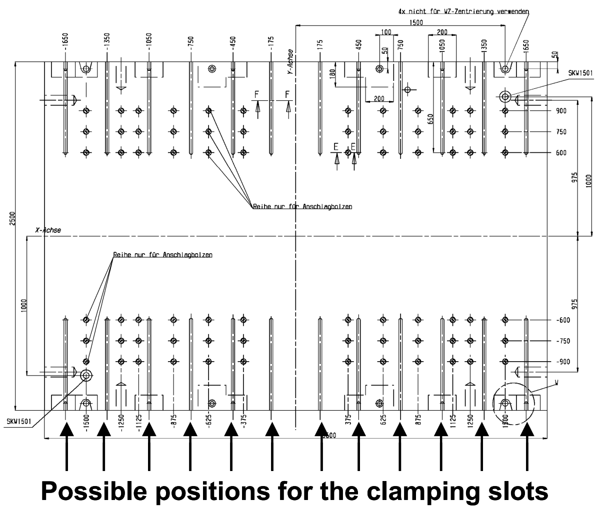


Figure 7-9: Top view of a press table.

Therefore the position of the clamping slots is the intersection of the axis of one of the press table's grooves and the side face of the lower die. Thus both positioning elements have to be integrated in the skeleton: a line depicts the groove axis and a plane is used for the side face of the lower die.

As soon as the position of the clamping slot is defined by selecting the inputs, its orientation remains to be defined. This is done automatically by interpreting the context. In fact, the horizontal orientation depends upon the topology of the input solid: the feature has to be placed so that its notch is oriented to the outside of the input solid and so that it is placed on the upper face of the solid in the normal axis. The positioning of the clamping slot is depicted in Figure 7-10.

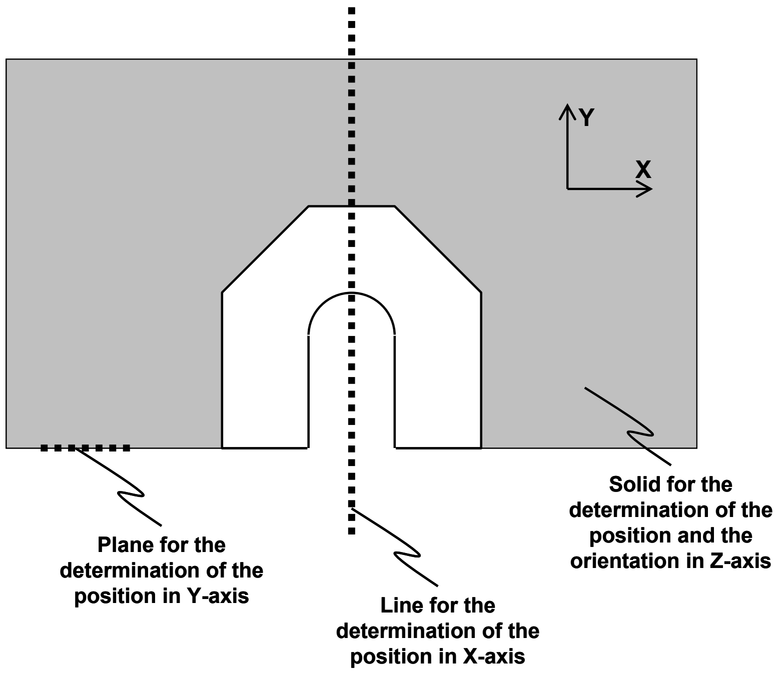


Figure 7-10: Clamping slot positioning.

Parameterization

The width of the press table grooves may have a varying value. Depending upon this value, the clamping slot dimensions have to be adapted: the width and the length of the notch may vary as well as the main width of the features and the bolt used for the fastening. All these dimensions have been extracted from the standards and have been directly linked to the press information by means of rules, allowing the clamping slot dimensions to be automatically adapted to the press characteristics.

Process information

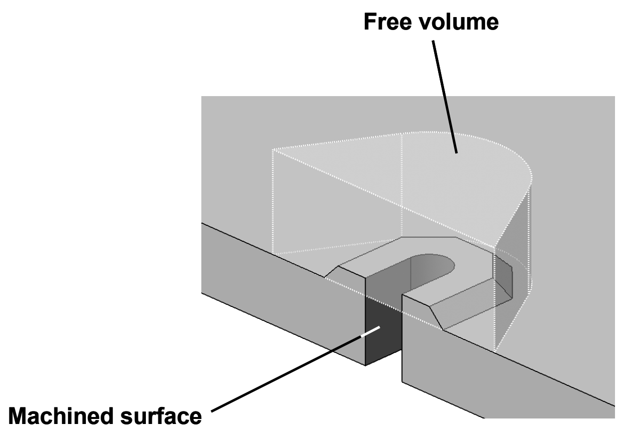


Figure 7-11: Visual representation of process information.

Apart from the dimensions of the clamping slot itself, the standards define the volume surrounding it that is to be kept free to allow the mounting of the stamping tool. This free space is required for the motion of the screwdriver. As a consequence, designers have to keep in mind that no geometry can be inserted into this volume. Traditionally, compliance with this rule is checked after completion of the design. Thus, although errors can be detected and corrected, they cannot be avoided. That is why the free volume has been integrated into the definition of the feature and may be used permanently for the detection of collision with other elements.

In this phase, it is also necessary to check the number of the clamping slots. This was done by visualizing the model. This operation can be avoided by taking advantage of the features that can be easily identified in the structure tree by a macro programmed in CATScript language (internal CATIA V5 programming language based on Visual Basic).

As further process information, the surfaces to be machined according to the standards are identified by a particular color that informs the production engineer of the correct finishing quality. As is the case for other information, this is integrated into the feature definition so that designers do not have to insert it explicitly.

7.7 Detailed Model and Conclusion

The template shown in Figure 7-5 has been further specified to obtain the finished hood stamping tool model. This is not the task of engineering template developers but that of stamping tool designers. Thus the model as it was presented in the figure was the start-off point for designers. As Figure 7-12 indicates, the template has been adapted (ribs on the top of the matrix) and details in the form of integrated components (guiding column, etc.) and features (clamping slot, holes, etc.) have been inserted.

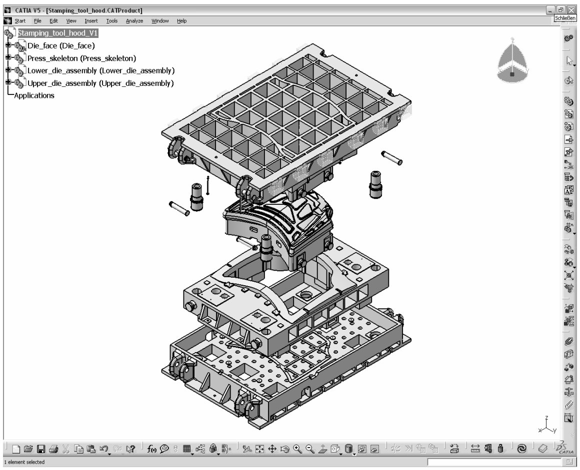


Figure 7-12: Detailed stamping tool model.

Tests have shown that it is possible to employ this template for other types of components as long as it has not been fully specified. Figure 7-13 depicts an example of the template used for the hood as applied to a car roof. Since the roof is bigger than the hood, the dimensions of the template have been automatically adjusted during the insertion of the inputs (die face, punch opening, etc.). The topology of some elements has been adapted as well (ribs, profile of the lower die side face, etc.). This proves that the templates are usable for different contexts, which was one of the original goals of this work. After having fully detailed a model, it is no longer possible to adapt it to another type of component since this would involve too many rectifications, which would not only cost time but would also be complicated. However, the template may easily be reused for another variant of the same component type. This example hence demonstrates the importance of the definition of the template detailing level but it also makes an additional statement: it is possible to reuse a detailed model if the new context does not differ too greatly. In this way, this scenario proves the statement from the section 5.6.3.2.

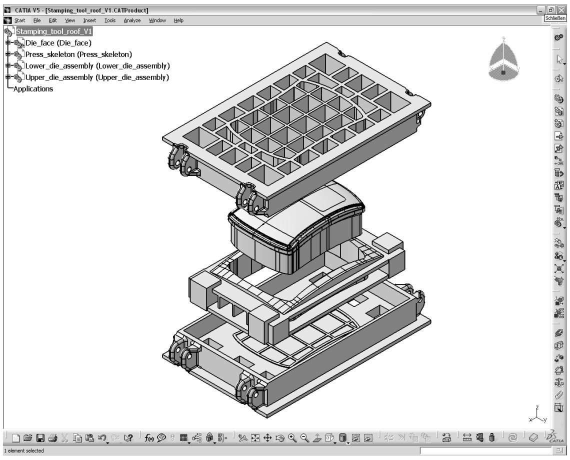


Figure 7-13: Stamping tool template applied to the roof model.

Chapter 8
Development Process

8.1 Introduction

In the previous chapter, the technical working of engineering templates and the associativity between models have been explained. The introduction of these new methodologies has an effect on the process run. As a consequence, the present chapter aims at defining the modifications of the stamping tool development process and evaluating the benefit generated.

8.2 Concurrent Engineering

8.2.1 Design Structure Matrix of Stamping Tool Development

To analyze and optimize the design process, we made use of the Design Structure Matrix (DSM) described in section 2.5.5 applied to the information collected in the second chapter of this thesis concerning the process chain analysis. Table 8-1 summarizes all the activities of the car-body and stamping tool development process. The activities are listed by level as was done in the SADT representation. In the last column, comments about each activity are given. They are based on the result of the last sections:

- Template: when the activity is supported by the template methodology
- Feature: when the activity is supported by the feature methodology
- Integrated component: when the activity is supported by the integrated component methodology
- Associativity: when the activity still exists but is automatically performed thanks to the associativity
- Macro: when the activity does not have to be performed by designers anymore and is supported by a macro
- Suppressed: when the activity is no longer necessary thanks to given improvements
- Unchanged: when the activity remains the same

Note 8-1
The activity "evaluate formability" has not been described in detail as it is fully performed in the simulation system and cannot be influenced.

Level 1	Level 2	Level 3	Level 4	Comment
Develop component	Model component	Model main shapes		Template
		Model form details		Features
		Model outline		Template
		Model holes		Features
	Evaluate formability			Unchanged
	Model die face	Position component		Associativity
		Fill holes		Suppressed
		Remove flanges		Suppressed
		Model blankholder		Associativity
		Model addendum	Lenghten surfaces	Suppressed
			Model addendum outine	Associativity
			Model addendum flange	Associativity
			Round surface	Associativity
	Manufacture prototype			Unchanged
	Test prototype			Unchanged
Develop stamping tool	Plan production facilities			Unchanged
	Design die face	Begin die face		Template
		Define method plan		Unchanged
		Evaluate formability		See above
		Model die face		See above
		Terminate die face		Suppressed
	Design stamping tool	Generate flow chart		Unchanged
		Model stamping tool	Prepare model	Suppressed
			Model main forms	Template
			Insert standard components	Integrated components
			Design details	Features
		Generate documentation		Macros
Produce stamping tool	Plan machining operations			Unchanged
	Cast components			Unchanged
	Machine components			Unchanged
	Order components			Unchanged
	Assemble stamping tool			Unchanged
	Constrol assembly			Unchanged

Table 8-1: List of the process activities.

This table has been used in a second step to generate the DSM. Here, only the last detailed level is displayed since it represents the elementary tasks. Suppressed activities are not displayed. Table 8-1 shows the result of this analysis with the relationships identified by a cross if they are standard relationships or by a T if a template is used in this context. This precision is important as templates consist of predefined information that is immediately available, being adapted later to obtain the definitive stand. This particularity hence has to be considered in the definition of the modeling process.

As the evaluation of the formability and the modeling of the die face will are with the template methodology allowing a full associativity between the simulation model and the die face model, both corresponding activities have been grouped in a single one (see Table 8-2).

		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
Model comp. main shapes	A	■				T		X													
Model comp. form details	B	T	■			X															
Model comp. outline	C	T		■		T															
Model comp. holes	D	T			■																
Evaluate formability/Die face	E	T		T		■				X											
Manufacture prototype	F		X	X		T	■														
Test prototype	G						X	■													
Plan production facilities	H								■												
Define method plan	I	T				X				■											
Generate flow chart	J	X							X	X	■										
Model tool main forms	K	T						X			T	■	X	X							
Insert standard components	L											T	■	X							
Model tool details	M											T		■							
Generate documentation	N											X	X	X	■						
Plan machining operations	O													X	X	■					
Cast components	P											X		X			■				
Machine components	Q															X	X	■			
Order components	R												X		X				■		
Assemble stamping tool	S																	X	X	■	X
Control assembly	T																			X	■

Table 8-2: Design structure matrix of the stamping tool development process.

8.2.2 Analysis of the Design Structure Matrix

The analysis of the matrix shows three cases of coupled activities: the loop from the formability evaluation to the component design, the loop from the prototype to the component design, and the loop from assembly control to the assembly. Second, four columns are more filled than the others and thus have more importance. They correspond to the following:

- The modeling of the component's main shapes
- The evaluation of the formability
- The modeling of the stamping tool's main shapes
- The design of the stamping tool's details

The modeling of the component's main shapes is the starting point of the current process and it will remain so as nothing can be done as long as no rough information about the component is available.

Up to now the formability simulation was carried out after the component design, but the matrix clearly demonstrates the importance of this activity and proves that it may begin earlier. That is why the formability simulation will be placed just after the modeling of the main shapes. This aims at enhancing the product quality and validating it as soon as possible by increasing the communication between both activities. Additionally, and to favor the iterations loops as long as they do not have too great an effect on the remainder of the process.

The definition of the flow chart and the modeling of the stamping tool's main shapes also play an important role. In fact, this is the basis for the rest of the geometry. This step should hence begin as soon as possible, as the first draft of the method plan is delivered after that. This step may also be moved forwards, after the initial evaluation of the formability. The result of both activities will, of course, be updated during the run of the process.

The fourth main identified activity is the stamping tool detail design. Its significance is justified by the fact that it is the last point of the stamping tool design and is necessary to produce the design. The detail design as well as the following steps (machining, assembly, etc.) might be done before the overall process is accomplished, for instance, if some of the components are fully specified. Yet generally this will not be possible because of the interferences between the components, so that this should be done all at the same time.

The definition of the component outline has more importance than the details and the hole definition since it is necessary for the design of the die face addendum. That is why this operation will be placed earlier in the process. The component's detail definition is necessary to achieve the design but it is not needed to create the first draft of the die face and the stamping tool geometry and may be done later.

In the DSM method, two kinds of situations should attract attention: empty rows or empty columns. An empty column means that the corresponding task does not provide any information to any other task and could be placed at the end of the matrix. This is the case for the activity "Model holes", which does not have any influence on the considered process. An empty row indicates that the corresponding activity does not need any information from another, for example, the task "Plan production facilities". This task can then be placed either at the beginning of the process or parallel to other steps. The result of the optimized matrix is showed in Table 8-3.

		H	A	E	C	I	J	K	F	G	B	L	M	N	O	P	Q	R	S	T	D
Plan production facilities	H	■																			
Model comp. main shapes	A		■	T						X											
Evaluate formability/Die face	E		T	■	T	X															
Model comp. outline	C		T	T	■																
Define method plan	I		T	X		■															
Generate flow chart	J	X	X			X	■														
Model tool main forms	K		T				T	■		X		X	X								
Manufacture prototype	F			T	X				■		X										
Test prototype	G								X	■											
Model comp. form details	B		T	X							■										
Insert standard components	L							T				■	X								
Model tool details	M							T					■								
Generate documentation	N							X				X	X	■							
Plan machining operations	O												X	X	■						
Cast components	P							X					X			■					
Machine components	Q														X	X	■				
Order components	R											X		X				■			
Assemble stamping tool	S																X	X	■	X	
Control assembly	T																		X	■	
Model comp. holes	D		T																		■

Table 8-3: Optimized design structure matrix.

8.2.3 Optimized Process

Normally the optimization of the DSM method should lead to a matrix where all the crosses are concentrated on the left-hand side of the diagonal line. Therefore activity G (Test prototype) should be moved to the left to avoid the isolated cross. This would be possible if only one prototype were produced after the first component draft. This is an additional argument that indicates the importance of the simulation, but unfortunately no assurances can be made that one prototype is sufficient. Consequently, this task is still necessary to validate the design.

Figure 8-1 shows the chronological arrangement of the new process, deduced from the DSM. The boxes in black represent the first draft of each activity, that is, the first information needed to start following activities, whereas the boxes in white represent the continuation of these activities, where information could be updated. The achievement of each activity is necessary to achieve related activities as well. Arrows represent information flows.

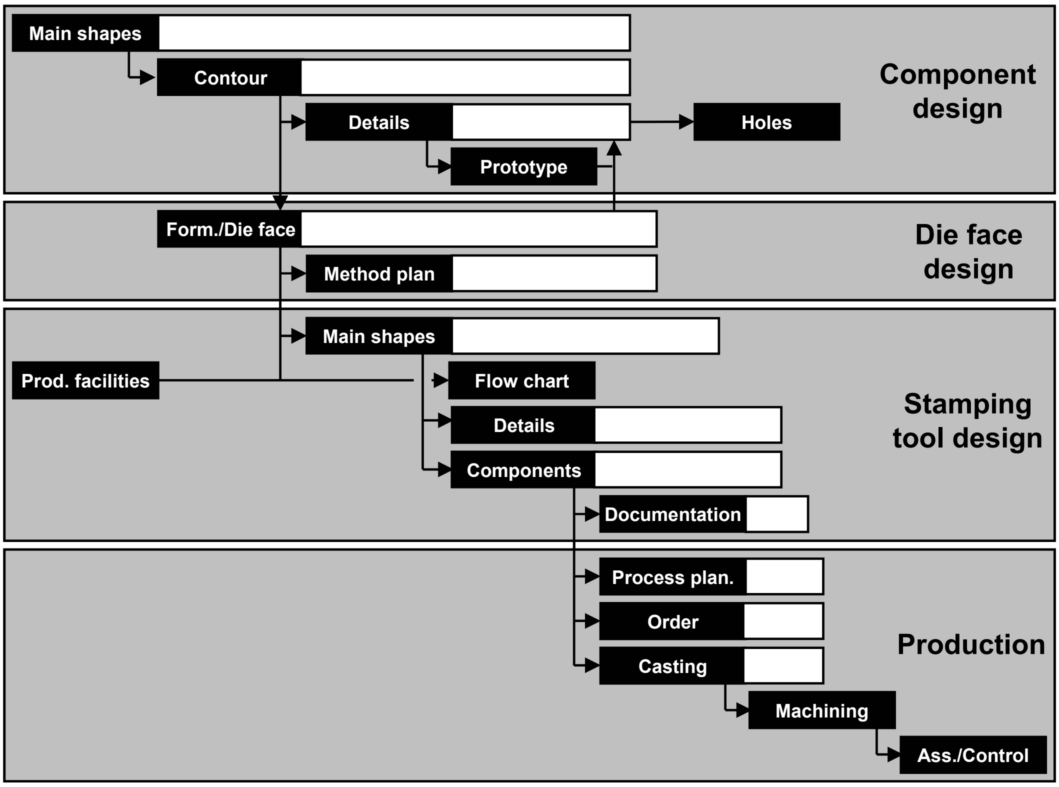


Figure 8-1: Optimized stamping tool development process.

To conclude, the following results have been reached:

- The result makes it clear that it is possible to start stamping tool design while the component is being optimized.
- The time of the single tasks is reduced thanks to functional elements.
- The global process time is reduced as each task starts earlier.

Independently of the duration of each step, the load of modeling work is reduced thanks to associativity and knowledge integration. As a consequence designers, who have to deal with more and more other tasks than the modeling, will have more time.

Nevertheless, some questions should be answered: as the simulation should take a bigger place in the process, who should do it? Should component designers be trained to perform the simulation task? Should the simulation be the first step of the stamping tool definition, or should it be integrated in the component development? But in some companies the formability simulation and the definition of the method planning are carried out by the same engineer so that it makes more sense to regroup the simulation not with the component design but with the die face design. Thus, either the cooperation between component and die designers should be intensified or the simulation could be done by different persons: component designers for the initial geometry validation and die face designers for the definitive validation and the die face design.

8.3 Product Data Management

8.3.1 Stamping Tool Engineering Templates and PDM Systems

In the process we are talking about, there are three kinds of templates to manage: car-body component, die face, and stamping tool templates. They are linked under the specifications mentioned Chapter 6 and Chapter 7. The three templates could be grouped in one entity, which would represent a project template. In the view of the PDM system, it would be represented as generic structure. If we were only considering the component and stamping tool design processes, the ideal way would be to start a new project on the basis of this generic structure. A preliminary task would be to generate a version from an existing structure containing the tree templates and to further specify them. However the problem is much more complex for the following reasons:

- Models are not involved in only one process chain. For example, the component model cannot be considered as belonging to the generic structure of the stamping tool as it is fundamentally created to define the car assembly and belongs to this structure
- Models have geometrical links to each other. This means that geometrical elements of different parts are linked to use a reference geometry of a model in another one.

Both problems are represented in Figure 8-2.

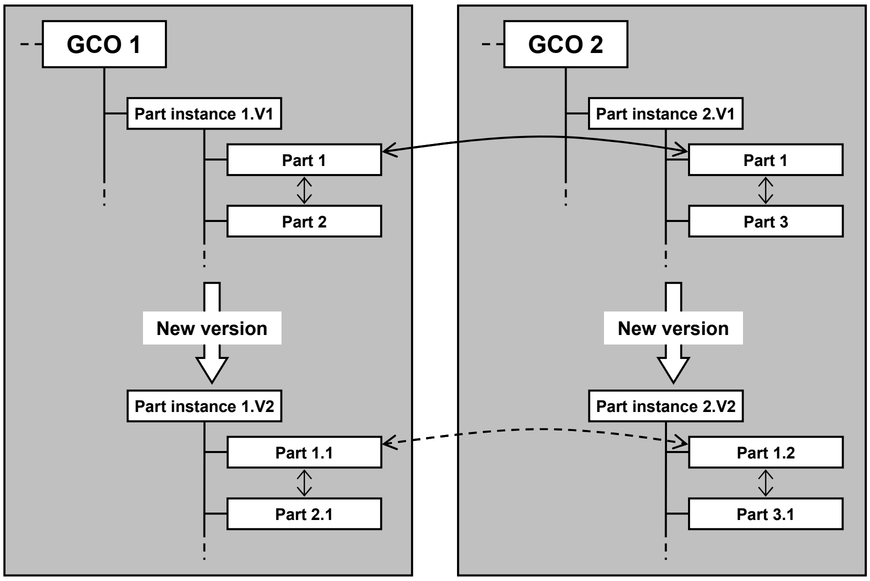


Figure 8-2: Creation of product variant and model interrelations.

Note 8-2
"GCO" means generic component in the PDM system. In this case, it is a generic structure on the assembly level with part instances that are also to be considered as assemblies. These terms should not be confused with the terms "integrated components" and "part" as employed in the remainder of this document.

Both generic components represent two structures that are different but contain a common part, namely Part 1. Each structure has its own specific part: Part 2 for GCO 1 and Part 3 for GCO 2. In the original state, Part 1 is the same; this means that it is inserted in both GCOs but represents one unique element. The link between both structures is represented by an arrow. Part instance 1 and Part instance 2 are design templates and are used to generate product models in the following way: new versions of Part instance 1 and Part instance 2 are created but without a link to the original part instances so that the parts are duplicated and can be modified or completed.

As a consequence of the duplication, versions of Part 1 are created and do not point to the same part anymore so that there is no equivalence between them and the associativity between the template instances is broken. To restore the link between the instances (represented by the stroke arrow), Part 1.2 should be replaced by Part 1.1. However, in this case, another problem arises: the geometrical links between elements of Part 1.2 and elements of other parts of the same structure would be broken.

Consequently, the use of the PDM system to derive product versions from a template generates restrictions for the template working and an alternative solution had to be found. To deal with it, the solution was to configure the product not in the PDM system but in the CAD system by assigning the necessary elements to their references in other parts. In the given example, this requires that Part 1.2 should not be replaced by Part 1.1, but geometrical elements within Part 1.2 should be linked to those of Part 1.1.

For instance, the GCOs could represent the die face and the stamping tool models. Then instead of inserting the die face model in the stamping tool structure, a different part should be prepared as buffer part and only elements of the die face model would be inserted in it such as the die face surface or the punch opening line. This means that it is not possible to modify them at this point; yet, thanks to their link to the original models, modifications are reflected from the die face model to the stamping tool model. Then these elements can be employed as inputs for the stamping tool modeling. There are efficient functionalities in CAD systems to create links between elements of various models and to replace these elements without losing the existing links. This method is depicted in Figure 8-3.

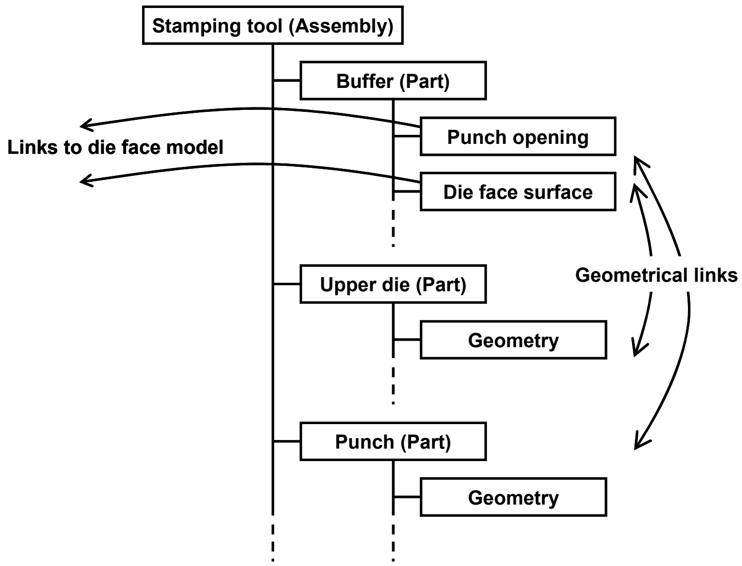


Figure 8-3: Use of a buffer part for the model linking.

8.3.2 Integrated Components and Features in PDM Systems

There are several methods in CAD systems to build a model on the basis of user-defined objects. Nevertheless if a PDM system is employed for the development process, the product structure and configuration are not done in the CAD system but in the PDM system. It should be defined how functional elements can be employed within a PDM system. Yet this is not only a problem of methods but also of functionality.

Features do not modify the product structure. They can therefore be inserted in the model directly using functions of the CAD system. For integrated components, the problem is more complex as they modify the product structure on the assembly level but also the geometry of the product's parts. This means that it is not possible to insert integrated components into a model in the PDM system. Functions that enable insertion of such objects belong to the CAD system. So, in this case, the product structure and its geometry are modified simultaneously. Subsequently, the PDM system has to recognize the structure modification and consider it in its representation of the product.

8.3.3 Release Process

According to [EiSt-01], the release and modification process has to be considered with the goal to reduce development time. It is one of the functions of PDM systems to support this process. They enable the status of an element to be defined: for instance inactive, in work, in validation, approved. When using templates, it is necessary to define which degree of template maturity corresponds to a particular status. The release process is closely defined with reference to the process depicted in Figure 8-1.

Templates could provide information for downstream processes before the corresponding activity is completed, as information is available even if the activity has not yet begun. It is therefore theoretically possible to start the following activity before the previous one. But, except in a few cases, it does not make sense to proceed in this way: if the detailing of the template-based model has not started, no information specific to the project is available. Thus the initial data release of each activity is not done if the initial specification of the model has not been completed, as shown in the previously mentioned figure. After that, the release and modification process is quite the same as a normal process without a template and is contingent on the model effectiveness. This means that designers have also to freeze the models while they are working on them and to release them when they are satisfied. The advantage of templates here is that modifications are propagated more quickly from one model to another.

8.4 Implementation and Software Requirements

8.4.1 Formulation of the Problem

The scenario presented is based on three commercial systems: a CAD system for the design, a simulation system geared for the definition and simulation of car-body panel drawings, and a PDM system. Each of the systems has its own philosophy, features, performance, etc. As a consequence, the ideal process underlies restrictions and has to be adapted. New design methodologies cannot be successfully introduced if they do not consider implementation software. On the other hand, CAx systems also have to develop to meet the customer's requirements. The present section expresses this.

Specialization
Although the automobile industry has its favorite CAD systems, these are still general systems, that could be used for the design of any product. Therefore, even if they contain various modules for different applications, they are not sufficiently tailored to enable effective and efficient working.

The first difficulty when working with general software is that these products offer too many functions, which are not easy to grasp for users, particularly for designers having much experience with older systems. Second, despite the large range of features, the desired or needed ones are not necessarily available. Further, CAD systems are still geometry oriented and not function oriented. They provide a geometrical feature-based modeling but do not offer functional feature-based modeling.

Finally, there is no method support offered to users. Each function of the software is explained but the choice of the right one and the design methodology are not mentioned, placing the burden on the users to think over how they should work with the software. Experience with other automotive companies has shown that most of them have similar demands and that they are carrying out almost the same research about modeling methodologies in parallel so that a common definition of software requirements is useful.

System interface
Other systems, in contrast, are extremely specialized, for example, the simulation system used to define and validate the geometry of the die face. And, in this case, other problems arise: the communication and the data exchange between applications. Data delivered from the simulation system are available in the exchange format and initially lose a large part of their information. Also, the data quality requires remodeling in a CAD system.

8.4.2 Customization

8.4.2.1 Specification of the Problem

The specialization of software should meet the requirements not only of a sector of the industry but also of each company - and this just does not seem possible. That is why software should be easily customizable.

At present, it is possible to customize software but only with a huge investment in programming. Customization requires in-depth programming competence and is extremely time consuming. On the other hand, there are possibilities to create user features and to duplicate geometry (for the template use). Yet the problem is that they are not powerful enough to define a high-functionality template.

Thus, whereas the first solution is too complicated in terms of time and investment, the second one would yield a too simple result. What is needed is the option to generate user-defined objects with many more possibilities than the ones offered today. The introduction of the template methodology implies new roles, which will be handled by different persons: template developers and template users. In the same way, software should have two levels of usability. The key system requirements are listed in the following section.

8.4.2.2 First Level of Usability for Specialists

Engineering template development
The first level of usability is foreseen for method specialists to develop the engineering templates, components, and features. Developing such elements is already possible within today's software under various terms. In this section, we will call the functionality intended for the definition of reusable and parameterized elements in the CAD software, the "user-defined object method".

User-defined objects make it possible to define a given sequence of modeling and to use it as a reference for further deployments. This sequence depends on the modeling context that regroups the user-defined object inputs. For the insertion of user-defined objects in a new context, the inputs have to be of the same nature as in the original context, which restricts the user's possibilities. The problem of user-defined object methods is that they do not offer the flexibility of other standard functionalities.

As illustration, we look at the definition of a line with a standard functionality: the user has to open the line definition dialog box and has various options available by the choice of inputs. The user may define the line by selecting two points, by selecting a point and a direction plane, or by selecting a point and a surface, for example. In addition, if the necessary inputs are not available, the line may be defined directly using the dialog box. Working with user-defined objects is quite different, as the nature of the inputs cannot be chosen as flexibly as mentioned. Moreover, they have to be prepared in advance. Users hence have to know exactly which inputs are needed and how the user-defined objects work before they insert one. That is why it should be possible for method specialists to define all the permissible user-defined object insertion routines so that end users could employ them in the same way as standard objects.

Parameterization problem
By defining the user-defined object parameterization and rules, developers have to choose which elements are masters and which ones are slaves. Thus the users' possibilities are restricted and it is difficult for them to understand the logical relations within the object. Indeed the definition of the rules managing the parameter values is unidirectional. This means that if a parameter A determines the value of a parameter B by means of a rule, it is only possible to set the value of A and not that of B. More flexibility in this domain is necessary to enhance the user-defined object usability. It would hence be better if developers were allowed to define bidirectional rules and to determine rule priority.

Topology and structure
The problem of topology is almost the same as that of parameterization. CAD software creates a hierarchy between geometrical elements, depending on the place they are located in the structure tree. Most of the time, the place of the elements is contingent on the order they are inserted in the model: thus the influence of the elements on the rest of the model can vary. Concretely this means that empty space could be undesirably filled by the insertion of further geometry or, vice versa, volumes that have to be kept intact might be removed. However, users should have the liberty to insert elements in the desired order they, making it necessary to manage the topology of the model.

To deal with this problem, we suggest two solutions:

- The order the elements are to be inserted is not known in advance but their desired position in the structure tree could be anticipated so that the element's position should be definable as standard properties. In this way, as the structure of the model is automatically met, the model topology is respected.
- Predefine the element's priority so that, in the case of a conflict between volumes to be added and volumes to be removed, the element having the higher priority will determine the result. For example, holes have the highest priority for the definition of removed volume and no positive volume should be added in their area even if inserted afterwards.

Personalized interface

In the currently deployed CAD software, modules can be personalized by adding tool bars and defining the functions belonging to them. The personalization options should be extended to the customization of a whole module intended for a specific domain of application. In this tailored GUI, all the developed user-defined objects have to be available.

It is also necessary that users have contextual help for employing user-defined objects at their disposal, as is the case for any standard function. Thus the creation of help documentation should belong to the definition of user-defined objects. Similarly, annex information should be linked to the user-defined object, for example, standards information or cost information. This should be accessible from the CAD model, but on the contrary to what can is currently the case, it should not be saved in the model itself. Information should be regrouped externally, first to limit the model size and second to be updatable centrally. This information may be called by the users when they need it but may also be prompted by the system when necessary, for example, to inform users about the consequence of their decisions.

8.4.2.3 Second Level of Usability for End Users

Discussion with a software provider has shown that some desired features would not be implemented as they could be too complicated for some users. Yet, the described solution for the user-defined object development is necessary although it may require particular skills. That is why a second level of usability would be expected.

The second level of usability belongs to the end users, namely designers. Their tasks should be facilitated by the previous work of developers. For them, the modeling philosophy based on user-defined objects should not be more complicated than the use of other functionalities. In addition, the personalized interface depicted above should make all the necessary information available directly from the CAD model to avoid time spent in the information search.

8.4.3 System Integration

During the building of the scenario, several problems appeared and were caused by systems interferences. Indeed the development and utilization of user-defined objects are relative efficient as long as this is done within the CAD system. However, the use of a PDM system imposes restrictions for the modification of the product structure that might conflict with the CAD functionalities, in particular, due to the fact that the PDM system creates generic structures that are different from the assembly representation in the CAD system so that some functions are no longer employable. This is the case of the integrated components that could not be inserted in such a generic structure.

As mentioned above, it is necessary to attach information to given elements within a model, for example, help files, context information, or directives. This kind of information could be specific to a user-defined object type or specific to an element in particular. That is why this differentiation should be supported by the PDM system, which should provide the access to the relevant information. Today's PDM system functionalities are not sufficient in the sense that they only allow information data to be attached to a component model but do not make it possible to attach information to an element within a model. Moreover, information would be managed by PDM systems, but directly accessible from the CAD system. This also applies for the user object libraries that are to be synchronized.

8.5 Conclusion

This chapter has depicted an important result of this thesis: it has shown the possibilities offered by functional elements to reduce process chain time. In particular, engineering templates make it possible to anticipate the beginning of a task and to reduce the design time and the modification time. That is why the process of the stamping tool design could be reorganized and is now in line with the concept of concurrent engineering.

The motivation to develop new design methods was partially justified by the evolution of the software possibilities. Paradoxically, one of the results of this chapter is the establishment that these are not sufficient to fully support the engineering template methodology. In particular, the archival of templates and the derivation of variants could lead to the breaking of the links in the case of several templates involved in different structures.

Finally, working with engineering templates modifies the habits of designers and calls for new skills. That is why this aspect has to be taken into account right in the template conception phase. The presented solution consists of introducing new roles corresponding to template developers and template users.

Chapter 9
Conclusion

9.1 Contribution

9.1.1 Outgoing Situation

On one hand, CAx software is evolving continuously, offering more and more features and flexibility in its deployment. On the other hand, there is a constant pressure in the companies to reduce the development times and to improve product quality. Therefore companies should not only adapt themselves to progressive developments in software, enterprises also have to reap benefits from them in order to be more competitive. Currently, the software evolution mentioned is going in the direction of complexity, although usability and ergonomic properties have been enhanced. In fact, the issues now of interest do not focus on the use of single functions but on working with the overall system because of the increasing number of possibilities opened.

Hence, new methodologies are needed for the design process in which the aim is not only to understand how systems work but rather to adapt the systems deployed to their own needs. These methodologies have to be much more detailed than in the past as, unlike previous modeling systems, the new ones do not deliver a static result without consideration of the way it is obtained - they additionally provide the possibility to structure the models, to parameterize them, to display the design history, and to link information.

9.1.2 Synopsis

Part I
The literature analysis in the first part of this thesis has depicted the possibilities offered today for the development process in terms of technologies and methods. In summary, there are ideas being developed for many years now that still have interesting application possibilities and that indicate the direction that should be followed: features are a first contribution underlying the importance of application specific solutions and continuous process information. Then there are other methods grouped under the term "reusable models" that give a first appreciation about what could be done for the acquisition and reuse of knowledge. Apart from knowledge-based systems, there are methodologies consisting of inserting information directly into the models or linking information to them. Although many kinds of information could be necessary in the development process, most of the time work centers on the formalization of manufacturing and cost information. Furthermore the integration of information in a model often concerns only a part of its elements and the linking of information between the models is also only partially realized.

Another form of knowledge integration is the development of active models: models or parts of the models can be defined in order to react to the context or to the designers' intents. This is realized by model parameterization and the incorporation of rules that take on a part of the designers' tasks. This reduces designer effort and cuts down on routine jobs.

Simulation of any kind is an increasingly important topic. Although it is employed in many stages of the process chain, two comments are called for: simulation takes place late in the processes so that it has only the role of validation and not really of development and optimization of the product properties. Further, there are separations between simulation systems and the other systems deployed, resulting in low flexibility in their joint utilization.

The topic of concurrent engineering has also been treated in the first part since it is considered as a certain evolution of development processes. Yet, although its principle is well known and has been described in many documents, concrete implementations and supporting methodologies in the application level are missing. Nevertheless product data management systems are an available solution for the management of data and the support of collaborative working.

Many of the contributions presented in the first part are treated without reflection on the adjacent requirements so that they deliver solutions for isolated problems. Because of the need for global solutions, it is also necessary to have a holistic view of the process and to pay particular attention to the interdependencies and interworkings of the processes.

The second chapter of the first part provides an example of an actual development process, the stamping tool development process. This chapter aimed at portraying the current situation and comparing it with the research results mentioned above. Thus it was possible to identify the implementation degree of new technologies and also to identify the gaps in the process and to define the goal of the present work. We found that many innovations available today were not transferred into practice and that there are consequently improvement potentials in the current design process. Moreover, this section set out a real example that could be used later for the scenario validating the developed solution.

The stamping tool development process is an established one, having been developed over years. In fact, the technical principles of sheet metal production are no longer evolving rapidly so that the key innovation potentials are not found in the product itself but in the definition process. In the considered process, the following points can be noted:

- There is no systematic reuse of existing information from previous projects: information is used only implicitly by designers.
- There are many isolated and different information sources to consult.
- There are many repetitive tasks from project to project but also in the same one.
- There is not enough collaboration between the various disciplines and no technical support for it.
- The whole process is quite static. This means that the planning and the information flows are previously defined and there are not many possibilities to deviate from them.
- Downstream model validations using simulation or prototypes may lead to expensive design modifications.

- The recent introduction of new CAD systems stimulates people to pay more attention to the definition and standardization of modeling methods.

Part II

In the second part of the thesis, a solution is described to enhance product development process efficiency. It consists of functional elements being developed for the specific engineering domains where they are employed and having the corresponding properties. The advantage of functional elements is that they fulfill different requirements: they improve the design methodologies in each step of the process chain but also guarantee a continuous information flow between the steps.

Consequently, functional elements are depicted as a central object for the product development process and have to be adapted to several influences: first, functional elements are intended for designers who are the first users. For designers, functional elements should replace low-level entities of CAD systems, which are typically employed in the current processes. They are central elements for designers and should form the interface between users and necessary information.

Information relating to designers' know-how belongs to functional elements. It depicts their behavior in given situations such as their insertion in a model, their updating, and their interactions with each other. In addition to this kind of information, other information derives from standards or belongs to adjacent steps of the process chain. While such information is not necessarily significant for product designers, it is automatically set to speed up the process.

CAx systems have to be taken into account for the definition and the implementation of functional elements since they strongly affect their behavior. Indeed functional elements take on part of the designers' routine jobs and thus their working has to be adapted to the designers' intentions on one hand and to software functionalities on the other hand. The second constraint is the associativity between functional elements in different systems. These elements have to be developed so that their various definitions are the same for the users although systems do not work in the same way.

Functional elements have been subdivided into three categories: engineering templates, functional features, and integrated components. Engineering templates are defined as reusable and knowledge-based elements on the component or assembly level. They do not define the overall geometry but are parameterized so that they describe a large part of the product geometry. Their contribution lies in the fact that they automate the tasks with little or no innovation potential, permitting designers to concentrate on the important aspects of their work. The remaining product attributes are defined conjointly with functional features and integrated components.

Functional features as they are presented in this thesis are first optimized for the design stage. They do not force designers to think like production engineers but have a behavior adapted to their function. The setting of information for production planning is then done in the background. Integrated components are inserted in models at the assembly level and simultaneously at the component level. They are based on inserted components but also modify other components by integrating surrounding elements necessary for their working.

The three functional elements presented are not developed separately but are brought together using a skeleton belonging for the most part to the template. In this way, new processes profit from a greater degree of standardization, which is a benefit for concurrent engineering. Indeed it enables a better comprehension of the data and an optimized access to necessary information by structuring the models and defining the task sequencing. This function is then supported by a PDM system whose aims are, in this context, to generate instances of the template, to enable links between different templates, to provide libraries for the rest of the elements, and to facilitate data exchanges.

Part III

As application for the methodologies described in the second part, the case of the stamping tool design development process has been chosen. The particularity of the car body process chain is that the production is not directly organized around the product data (the car-body component) but is done over the development of a stamping tool. Thus the various models to consider are a good example to demonstrate the utility of templates and the way several templates could work together.

As a consequence, the component model, which is the first one in the process, has to be developed with the template methodology and optimized not only for immediate use but also for its further use as input for the die face template. Both component templates and die face templates integrate simulation information to anticipate the result of the method plan and to create an associative link between the model of the die face in the CAD system and the same model in the simulation system.

In the same way, a stamping tool template has been developed and associated to the die face template so that, in the end, the associativity is insured from the component model up to the stamping tool model. This means that setting and modifying the component parameters automatically involve adaptations of the stamping tool parameters. Based on the associativity, information for following steps is available earlier. Tasks can hence be started before downstream ones are completed and their detailing is done in parallel. This principle of simultaneous engineering necessitates a revision of the actual process, leading to a reduction in the overall process duration and to greater flexibility for engineers. The template methodology and the model associativity do not suppress iteration loops in the development process. On the contrary, they favor them by simplifying the change management. Small improvement loops such as that between die face design and simulation are promoted in comparison to larger ones, which call for more effort and are much more expensive.

This scenario has demonstrated that it is possible to use templates twice. First, a template has been developed for a generic single-action stamping tool. Then it has been employed for the case of a car hood and has been further detailed so that it is now a specific template that can be used for other hood variant. In the same manner, the base template could be employed for the generation of specific templates of another kind.

Finally the development of the scenario has shown that software already includes some helpful functions to support the creation and utilization of user-defined objects. Yet there are still potentials for improvement in this direction. It has also been made clear that the user is not to be underestimated: users are a key factor for the success of the implementation of new methods.

9.1.3 Added Value

The solution developed in this thesis bases on functional elements and especially on engineering templates. Their introduction enables and necessitates new organization. It "enables" a new organization in the sense that, thanks to templates, collaborative engineering can be intensified. But it also "necessitates" a new organization since the working with templates is different from before. Indeed before they are employed in the design process, templates have to be developed and maintained, which requires establishing new roles. Further, the designer's tasks are also modified so that designers should learn the new methodologies and take part in trainings.

The compensation for the adaptation effort required from the designers is that they have methodologies and technologies at their disposal which are much more intuitive to employ and are better adapted to their needs. Moreover the frequency of repetitive tasks has been reduced; the access to relevant information and the evaluation of the design as well as cooperation have been simplified.

In the view of the process, more flexibility has been introduced thanks to the possibility to work with incomplete data sets and to easily update them later. For the same reason, the duration of each task has been reduced and their starting point advanced. Thus, the duration of the overall process has been shortened.

Model quality has been enhanced by the introduction of process information, active behavior, and model links. The consequence is improved product quality as well - namely, better models and new methods support the users in product development, helping them in decision making and taking on a part of their work by means of automation. Designers can hence concentrate on the most important part of their job: innovation. The fact that engineering templates already define a large part of the product enables designers to invest more time in the essential points.

9.2 Outlook

As mentioned throughout this thesis, it is important to adapt the design methodologies to the user requirements in each domain of engineering. Templates, integrated components, and functional features are a first step in this direction, but the development of specific solutions should be extended and it would also be necessary to create custom interfaces for user-defined object utilization. In fact, although templates are specific to an object, the functions geared for managing them in the CAD software remain general and could be improved. Thus an entire platform intended for stamping tool design with all the necessary functionalities and information could be created on the basis of the existing CAx systems.

The core topic of this dissertation was the development of design methods for stamping tools yet the ideas described here could be generalized for two reasons. First, reusable knowledge-based elements are pertinent not only for stamping tool design but also for a variety of other domains in the automotive industry or in other sectors. Stamping tool design is a discipline where standardization reaches a high level, but it is surely possible to create engineering templates for other application areas than car-body components, the die face, and the stamping tool. In other cases, the requirements set for the templates, the level of definition, the related knowledge, etc. may be different yet the concept would be the same.

Second, the aim of the thesis was to consider all the possible aspects intervening in the considered process but the work had to be limited. And although the presented result is consistent in the defined frame, the view of the process could be enlarged. Concretely, the stamping tool design process is not a closed process and, to be comprehensive, the adjacent processes would have to be considered as well. That means that a comparable survey should be done at a higher level, as indicated in Figure 9-1. In particular, the component design requirements have been envisaged only from the view of the stamping tool, but many more have to be treated: for example, the requirements for the hole trimming tool or for the inspection planning. Concurrent engineering has been introduced in the stamping tool development process but this is only a small part of the global process that should also be organized in a concurrent way.

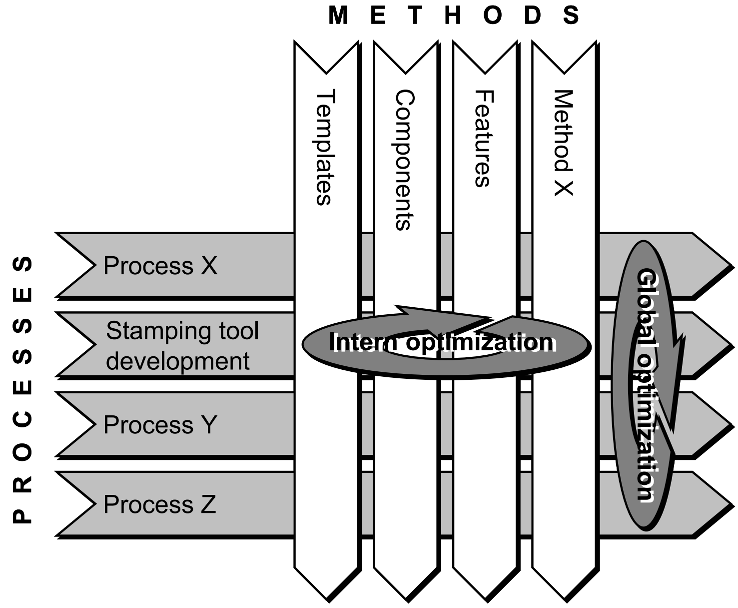


Figure 9-1: Extension of the analysis to the overall process.

In this study, the integration of simulation information in templates has been demonstrated using the example of the formability simulation but there are other kinds of simulations that should also be taken into account. This is particularly important because of the increasing significance of this discipline.

Further, various templates and their relationships have been described; yet the issue at hand is that of design templates. If necessary, it is certainly possible to create other forms of templates and to link them together. For instance, it would be interesting to introduce production planning templates and to match their attributes with those of the design templates.

Nowadays it is not possible to introduce product development methods without taking all the systems belonging to the process into account. That is why the engineering template methodology can be implemented only if it is developed on the basis of CAD and PDM systems jointly. Improvements are still necessary in this area as PDM systems do not provide all the expected functionality for the derivation of models from a master model considered as template.

Bibliography

[AlFa-97] T. Altan, P. Fallböhmer, "Entwicklungen im Werkzeugbau in Forschung und Industrie – Perspektiven aus den USA", Proceedings of the EMO, Hannover, 15-16 September 1997

[AnKa-94] B. N. K. Ann, C.C. Kai, "A knowledge-based system for strip layout design", Computer in industry No. 25, 1994, pp. 31-44

[BäHa-01] T. Bär, S. Haasis, "Verkürzung der Entwicklungszeiten durch den Einsatz von Skelettmodellen und der Feature-Technologie", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001. pp. 143-155

[Barn-53] H.G. Barnett, "Innovation: The basis of cultural change", McGraw-Hill Book Company, 1953

[Beck-03] T. Beck, "Automotive electronic systems – Beherrschung der Komplexität", 7. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2003

[Besd-99] D. Besdo, "Zur Problematik der Vorhersage von Rückfederungen bei Blech-Umformsimulation", Virtuelle Umformtechnick, Festschrift zu Ehren von Professor Kurt Lange anläßlich seines 80. Geburtstages, Umformtechnick 2000 Plus, 1999

[BeSt-96] S. Berndes, A. Stanke, J. Piller and U. Wagner, "A concept for revitalization of product development", Concurrent simultaneous engineering systems – The way to successful product development, Springer 1996

[BiBr-00] R. Bidarra, W.F. Bronsvoort, "Semantic feature modelling", Computer aided design, No. 32, 2000, pp. 201-225

[BiTe-93] R. Bidarra, J.C. Teixeira, "Intelligent form feature interaction management in a cellular modeling scheme", Proceedings on the second ACM symposium on Solid modeling and applications, Montreal, Canada, 1993, pp. 483-485

[BNBH-01] W.F. Bronsvoort, A. Noort, J. van den Berg, G.F.M. Hoek, "Product development with multiple-view feature modelling", Feature based product life-cycle modelling, Conference on feature modelling and advanced design-for-life-cycle systems, Valenciennes, France, 12-14 June 2001

[BoAo-03] C. Bouchard, A. Aoussat, "Modellization of the car design process", International journal of vehicle design, Vol. 31, No. 1, 2003

[BoGi-01] J.C. Borg, F. Giannini, "Exploiting integrated product & life-phase features", Feature based product life-cycle modelling, Conference on feature modelling and advanced design-for-life-cycle systems, Valenciennes, France, 12-14 June 2001

[Brow-04] D.H. Brown associates inc., "Design creation and validation, key research topics and solutions", Product lifecycle management, 2004

[BrTi-00a] D. Brissaud, S. Tichkiewitch, "Innovation and manufacturability analysis in an integrated design context", Computer in industry, No. 43, 2000, pp. 111-121

[CKCK-00] J.C. Choi, C. Kim, Y. Choi, J.H. Kim, J.H. Park, "An integrated design and CAPP system for deep drawing or blanking products", Advanced Manufacturing Technology Vol. 16, No. 1, 2000, pp. 803-813

[Coll-86] "Collins dictionary of the English language", Second edition, Collins London & Glasgow, 1986

[CRMW-02] A. Cui, A. Riitahuhta, T. Ma, L. Wang, "tudy on integration of CAD and CAM in autobody panel", 12th International conference on flexible automation and interlligent manufacturing, Dresden, Germany, 2002, pp. 1258-1264

[CrPS-04] I. Crnkovic, A. Persson Dahlkvist, D. Svensson, "Managing complex systems – Challenges for PDM and SCM", Projects & Profits, No. IV-7, July 2004, pp. 36-42

[CuDi-88] J.J. Cunningham, J.R. Dixon, "Designing with features: the origin of features", Computer in engineering, Vol. 1, 1988

[Dank-00] C.W. Dankwort, " Present-day demands versus long term open CAD/CAM strategies", Process chain in automotive industry, 2000

[DeFG-96] T. De Martino, B. Falcidieno, F. Giannini, "Feature-based modeling as solution for concurrent engineering design", Computer Integrated Manufacturing and Automation Technology, Grenoble, France, 29-31 May 1996

[Dieh-98] L. Diehl, "Machining molds and dies", Manufacturing engineering, April 1998, pp. 34-37

[DFGH-94] T. De Martino, B. Falcidieno, F. Giannini, S. Haasinger, J. Ovtcharova, "Feature-based modelling by integrating design and recognition approach", Computer-aided design Vol. 26 No. 8, August 1994

[DiTT-01] P. Dietz, M. Tawil, D. Trenke, "3D-Produktmodellierung in der Entwicklungskette – Aufwand und Nutzen", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, pp. 113-130

[DiYu-02] L. Ding, Y. Yue, "An intelligent Hybrid approach for design-by-features", International conference on Information Visualisation, London, July 2002

[DiYA-00] L. Ding, Y. Yue, K. Ahemt, "An integrated approach to integrating CAD/CAM", Proceding of the 6th annual conference of the Chinese automation and computer society in the UK, Longhborough, 23-24 September 2000

[DKLM-97] C.W. Dankwort, P. Kellner, D. Leu, G. Müller, W. Renz, G. Weißberger, "CAD in der Automobilindustrie: Aktuelle Bedürfnisse und mittelfristige Ziele"CAD-CAM Report, No. 1, 1997

[Dürh-02] W. Dürheimer, "Mechanik Technologien – immer noch für modern High-Tech Automobile", 6. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2002

[EiSt-01] M. Eigner, R. Stelzer, "Produktdatenmanagement-Systeme – Ein Leitfaden für Product Development und Life Cycle Management", Springer 2001

[Erdr-01] K. Erdrich, "Mehr Potenzial für die Produktentwicklung", CAD-CAM Report, No. 2, 2001

[FeKr-03] J. Fetzer, S. Krainc, "Das Rad nicht neue erfinden", Form + Werkzeug, No. 1, 2003

[Fetz-00] J. Fetzer, "Feature-basierende CAD/CAM-Prozesse, Teil 1", CAD-CAM Report, No. 12, 2000

[Fetz-01] J. Fetzer, "Feature-basierende CAD/CAM-Prozesse, Teil 2", CAD-CAM Report, No. 2, 2001

[FFHL-01a] M. Frenzel, D. Frank, S. Haasis, A. Layer, H. Schulze, "Featurebasierte Kostenkalkulation in der Produktentwicklung", VDI 143 Produktentwicklung, No. 3, March 2001

[FFHL-01b] M. Frenzel, D. Frank, S. Haasis, A. Layer, H. Schulze, "Lösungsansatz für Feature- und Web-basierte Kostenkalkulation am Beispiel Betriebsmittelbau", Werkstatttechnik, No. 91, 2001, pp. 349-351

[FIHS-04] S. Florek, H. Horn, P. Schel, "System- und Prozessintegration als Schlüsselfaktoren", CAD-CAM Report, No. 10, October 2004

[FISP-93] Federal information processing standards publication, "Announcing the standard for integration definition for function modeling (IDEF0)", 1993

[FoGM-99] M. Fontana, F. Giannini, M. Meirana, "A Free Form Feature Taxonomy", Eurographics '99 / P. Brunet and R. Scopigno, Vol. 18, No. 3, 1999

[GeWe-01] H. Gerken, M. Weyrich, "Management von Erfahrungen mit einem Assistenzsystem für die CAD/CAM Prozeßkette", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, pp. 233-251

[Gill-01] T. Gill, "Managing intellectual property stored in CAD databases", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, pp. 53-57

[GPSM-01] B. Göllnitz, K. Priebe, A. Schreiber, T. Mechlinski, "PDM data exchange between Windchill and ENOVIA vpm at Webasto", User report, No. 2, 2001

[Guna-98] A. Gunasekaran, "Concurrent engineering: A competitive strategy for process industries", Journal of the operational research society, Vo. 49, No. 7, 1998

[Hara-04] M. Hara, "Moving target: the inventor of lean manufacturing continues to improve the process", Automotive Industries, June 2004

[Hart-90] J.R. Hartley, "Simultaneous engineering: The management guide", Industrial newsletters Ltd, Dunstable, UK, 1990

[Hart-92] J.R. Hartley, "Concurrent engineering, shortening lead times, raising quality, and lowering costs", Cambridge, MA, Productivity Press, 1992

[HFRW-01] S. Haasis, D. Frank, B. Rommel, M. Weyrich, "Feature-based integration of product, process and resources", Feature based product life-cycle modelling, Conference on feature modelling and advanced design-for-life-cycle systems, Valenciennes, France, 12-14 June 2001

[Hoch-03] R. Hochgeladen, "Objektorientierte Modellierung mit 3D-CAD-Systemen", CAD-CAM Report, No. 9, 2003

[Hoch-04] R. Hochgeladen, "Entwurf komplexer Zusammenbauten mit UML", CAD-CAM No. 3, 2004

[JHan-96] J.H. Han, "Survey of feature research", Technical report IRIS 96-346, Institute for robotics and intelligent systems, USC, USA, 1996

[KaSf-01] G.J. Kaisarlis, M.M. Sfantsikopoulos, "Feature-based process conversion – A knowledge based system approach", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, 203-215

[Kess-96] S. Kessler, "Enabling technologies I: The CONSENS platform", Concurrent simultaneous engineering systems – The way to successful product development, Springer 1996

[Klei-96] R. Klein, "G-Rep: Geometry and feature representation for an integration with knowledge based systems", IFIP 5.2 Workshop on geometric modelling in CAD, Airlie, Virginia, May 1996

[Klei-97] R. Klein, "Towards knowledge based feature modelling", 30th International Symposium on Automotive Technology and Automation, Florence, 16.-19 June 1997, pp. 207-215

[Kloc-02] F. Klocke, "Es gibt noch viel zu tun - In der CAD/CAM/NC-Kette steckt noch eine Menge Entwicklungspotenzial", Form + Werkzeug, November 2002

[Konst-00] "Standardisierung als Schlüssel zum Erfolg", Konstruktion No. 10, October 2000

[KrWe-99] F.L. Krause, C. Weber, "Features: Die neue VDI-Richtlinie 2218 schafft Klarheit", CAD-CAM Report No. 12, 1999

[KrRU-94] F.L. Krause, E. Rieger, A. Ulbrich, "Feature processing as kernel for integrated CAE systems", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[KrWö-01] F.L. Krause, T. Wöhler, "Automatisierte Ableitung von anwendungsspezifischen Sichten aus Konstruktionsdaten", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001. pp.373-387

[Kuma-03] V. Kumar, "Evolution of information technology and its emergence in India", Nistads News, Vol. 5, No. 2, October 2003

[LaHH-01] A. Layer, S. Haasis, F.J.A.M. van Houten, "Feature-based, design-concurrent cost calculation reasoning", Proceedings of DETC'01 ASME 2001, 9-12 September 2001, Pittsburgh, Pennsylvania, USA

[LaHH-02] A. Layer, S. Haasis, F.J.A.M. van Houten, "Case-based, design-concurrent cost calculation – Supporting the decision-making process by anticipating product costs", Intenational journal of advanced manufacturing systems, Vol. 5, Issue 1, 2002

[LaLM-94] K. Lagus, T. Laakko, M. Mäntylä, "ICONF – An incremental constraint facility in a feature modeling system", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[Laus-03] J. Lauscher, "Nutzenpotenzial in einer Feature-basierenden Prozesskette", CAD/CAM Technologie, N4, 2003

[Laye-03] A. Layer, "Cased-based cost estimation – A building block for product management and design-for-X", Ph.D. Thesis, 2003

[LiCh-96] Z.C. Lin, G.J. Chang, " Application of a neural network machine learning model in the selection system of sheet metal bending tooling ", Artificial intelligence in engineering No. 10, 1996, pp.21-37

[Lieb-03] B. Liebroth-Leden, "Beschaffung – Eine besondere Herausforderung für den Nischenhersteller", 7. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2003

[Lind-02] U. Lindemann, "CAx – Eine Kreativitätsbremse in der Produktentwicklung?", 6. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2002

[LSFX-03] Z.Z. Li, B. Song, Q.R. Fu, X.J. Xiao, "An integrated concurrent injection mold design approach", Concurrent engineering: the vision for the future generation in research and applications, 2003

[Luby-01a] S. Luby, "New tools complement CAD and PDM for detailed collaboration", Desktop engineering, September 2001

[Luby-01b] S. Luby, "Essentials of managing product design data", Machine design, 9 August 2001

[MbSH-02a] S. Mbang, H. Ströhle, S. Haasis, "Trends bei der Feature-basierenden Karosserieentwicklung", CAD-CAM Report No. 11, 2002

[MbSH-02b] S. Mbang, H. Ströhle, S. Haasis, "Intelligent feature model for flexible manufacturing", 12th international conference on flexible automation an intelligent manufacturing, July 2002, Oldenbourg

[MCOK-97] F. Mandorli, U. Cugini, H.E. Otto, F. Kimura, "Modeling with self validation features", Proceedings of the fourth symposium on solid modeling and applications, 14-16 May 1997, Atlanta, Georgia

[Mill-93] L.C.G. Miller, "Concurrent engineering design – Integrating the best practices for process improvement", 1st ed., Dearborn, MI, 1993

[Minn-91] S.L. Minneman, "The social construction of a technical reality: empirical studies of group engineering design", Ph.D. thesis, Department of mechanical engineering, Stanford University, USA, 1991

[MRSC-01] F. Mandorli, C. Rizzi, L. Susca, U. Cugini, "An approach to implement feature-based applications using knowledge aided engineering technology", Feature based product life-cycle modelling, Conference on feature modelling and advanced design-for-life-cycle systems, Valenciennes, France, 12-14 June 2001

[NaZa-01] E. Napalkov, V. Zars, "Identical transformations of design and processing structures for equivalent knowledge acquirement and reuse", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, pp. 361-371

[NoTi-04] F. Noël, S. Tichkiewitch, "Shared dynamic entities technology to support distant coordination in design activity", Annals of the CIRP, Vol. 53, No. 1, 2004, pp. 163-166

[OvJa-94] J. Ovtcharova, U. Jasnoch, "An integration of feature-based design and consistency management in CAD applications", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[Ovtc-01] J. Ovtcharova, "Einsatz von neuen Konzepten in der Produktentwicklung. Digital Mock-up: Das digital Engineering Backbone", Internationaler Rohbau-Expertenkreis, 7. Fachtagung "Prozesskette Karosserie", Fellbach, Germany, 22-23 February 2001

[OWVM-97] J. Ovtcharova, C. Weber, S. Vajna, U. Müller, "Neue Perspektiven für die Feature-basierte Modellierung", CAD/CAM No. 3, March 1997

[PaMa-01] N. Pavkovic, D. Marjanovic, "Considering and object oriented approach to the design process planning", International journal of technology management, No. 21, 2001

[PaWN-94] P. Palani, R.H. Wagoner, K. Narasimhan, "Intelligent design environment: A knowledge based simulation approach for sheet metal forming", Journal of materials processing technology Vol. 45, 1994, pp. 703-708

[Pete-98] J. Peters, "Supplier and buyer market power, appropriability and innovation activities – Evidence fort the german automotive industry", Volkswirtschaftliche Diskussionsreihe, Januar 1998

[Piet-00] M. Pietsch, "Karosserieteile virtuell herstellen – Prozesssimulation für die Automobilindustrie", Automotive circle international conference, 30-31 October 2000

[RATH-99] K. Roll, T. Altan, A.E. Tekkaya, M. Herrmann, "Virtuelle Umformtechnick, Festschrift zu Ehren von Professor Kurt Lange anläßlich seines 80. Geburtstages", Umformtechnick 2000 Plus, 1999

[ReZi-04] A. Rennet, M. Ziegler, "Datenmanagement im Betriebsmittel- und Vorrichtungsbau", CAD-CAM Report, No. 9, 2004

[RKFM-04] V. Rückel, A. Koch, K. Feldmann, H.. Meerkamn, "Prozessdatamanagement – der Schlüssel zum Erfolg", Zeitschrift für wirtschaftlichen Fabrikbetrieb, No. 10, 2004, pp. 582 - 586

[RoHe-99] K. Roll, M. Herrmann, "Determination of the influence of the forming process on the final product properties of sheet metal parts", Advanced Technology of Plasticity, vol. 3, Processing of the 6th ICTP, 19-24 September 1999

[RoHS-97] H. Rose, S. Haasis, H. Schulze, " Erfahrungsgeleitete Arbeit mit Prozeßketten in der Produktion", 1. Kassler Kolloquium, 1997, Kassel, pp. 486-492

[RoTi-00] L. Roucoules, S. Tichkiewitch, "CoDE: a co-operative design environment. A new generation of CAD systems", Concurrent engineering: Research and applications, Vol. 8, No. 4, 2000, pp263-280

[Rude-98] S. Rude, "Wissensbasiertes Konstruieren", Habilitation thesis, Aachen, Shaker-Verlag 1998

[SAFS-01] J. Schwarz, N. Adameck, D. Frank, R. Siebert, S. Haasis, "The use of feature-based workflow techniques in automotive product development", 7th International conference on concurrent enterprising, 27-29 June 2001, Bremen

[Sage-00] L.A. Sage, "Winning the innovation race – Lessons from the automotive industry's best companies", Ernst & Young LLP, 2000

[SBRU-94] J.J. Shah, G. Balakrishnan, M.T. Rogers, S.D. Urban, "Comparative study of procedural and declarative feature based geometric modeling", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[Schu-01] S. Schulte, "Blechumformsimulation im Produktentwicklungsprozess", 7. Fachtagung "Prozesskette Karosserie", Fellbach, Germany, 22-23 February 2001

[ScOt-04] M. Schaffrick, T. Otto, "Daten-Rückgrat für Concurrent Engineering", Digital Engineering Magazin, No. 6, October-November 2004

[Shah-91] J.J. Shah, "Assessment of features technology", Computer aided design Vol.23 n5, June 1991

[Sche-03] S. Scheermesser, "Messen und Bewerten von Geschäftsprozessen als operative Aufgabe des Qualitätsmanagements", Ph.D. thesis, Fakultät für Maschinenwesen der Rheinisch-Westfälischen Technischen Hochschule Aachen, March 2003

[ShMä-95] J.J. Shah, M. Mäntylä, "Parametric and feature-based CAD/CAM – Concepts, techniques and applications", Library of congress cataloging-in-publication data, 1995

[Sieg-99] K. Siegert, "Entwicklungstendenzen in der Blechumformung", Festschrift zu Ehren von Professor Kurt Lange anläßlich seines 80. Geburtstages, Umformtechnick 2000 Plus, 1999

[SiKA-91] S.K Sitaraman, G.L. Kinzel, T. Altan, "A knowledge based system for process-sequence design in axisymmetric sheet-metal forming", Journal of materials processing technology No. 25, 1991, pp. 247-271

[Slee-96] E. Sleeckx, "Enabling Technologies II, CONSENS Tools", Concurrent simultaneous engineering systems – The way to successful product development, Springer 1996

[SSJH-94] O.W. Salomons, F van Slooten, H.G. Jonker, F.J.A.M van Houten H.J.J. Kals, "Interactive feature definition", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[Stah-98] J. Stahl, "Entwicklung einer Methode zur integrierten Arbeitsgestaltung und Personalplanung im Rahmen von Concurrent Engineering", Shaker, 1998

[Stap-02] U.W. Stapelfeldt, "Innovation durch Standardisierung", Engineering business, February 2002

[Stoc-03] J. Stockmar, "Auswirkungen des technologischen Wandels", 7. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2003

[StTh-03] J. Stjepandic, M. Thel, "Datenaustausch: Form-Features und Parametrik", CAD-CAM Report, No. 2, 2003

[SVCA-94] A. Saucier, C. Vargas, P. Court, P. Albert, P.A. Yvars, "Compared application of two knowledge modelisation methodologies on a car engine cylinder head design problem", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[TeVG-02] S. Teleionis, G.C. Vosniakos, T. Giannakakis, "An expert aid system aid for technological decisions in planning sheet metal forming operations", 12th International conference on flexible automation and intelligent manufacturing, Dresden, Germany 2002

[TiBr-00] S. Tichkiewitch, D. Brissaud, "Co-ordination between product and process definitions in a concurrent engineering environment", Annals of the CIRP, Vol. 49, No. 1, 2000

[Tisz-95] M. Tisza, "Expert systems for metal forming", Journal of materials processing technology No. 53, 1995, pp. 423-432

[TiTo-93] S. Tichkiewitch, M. Tollenaere, "Toward integrated design process", COMES 93, 17-18 May 1993, Clermont-Ferrand, France

[ToBT-94] M. Tollenaere, P. Belloy, S. Tichkiewitch, "A part description model for the preliminary design", Advanced CAD/CAM systems, State-of-the-art and future trends in feature technology, 1994

[Tyso-01] R. Tyson, "Applying the design structure matrix to system decomposition and integration problems: A review and new directions", IEEE transactions of engineering management, Vol. 48, No. 3, 2001

[Uhly-04] V. Uhly, "Wettbewerbsvorteile durch innovative Produktionsstruktur und -technik", Internat. Produktionstechnisches Kolloquium, 2004, pp. 231-243

[UzRM-91] R. Uszoy, D. J. Ramcharam, L. A. Martin-Vega, "An experimental expert system for planning for sheet-metal parts" Computer and Industrial Engineering, Vol. 20, No. 1, 1991, pp. 59-69

[Walt-03] H. Waltl, "Veränderte Anforderungen an den Werkzeugbau im zukünftigen Produktentstehungsprozess", Automotive circle international conference, Bad Nauheim Frankfurt, 1-2 July 2003, pp. 169-184

[WaMi-00] F. Wang, J.J. Mills, "A product data model and processor supporting virtual product development", ASME 2000 Design engineering technical conferences and Computers and information in engineering conferences, Baltimore, Maryland, 10-13 September 2000

[WaSc-97] H. Waltl, S. Schulte, "Virtuelle Werkzeugentwicklung – Perspektiven der CAE-Techniken im Werkzeugbau", Proceedings of the EMO, Hannover, 15-16 September 1997

[WCFS-03] L. Wang, N. Cai, H.Y. Feng, W. Shen, "Feature-based reasoning for machining process sequencing in distributed process planning", Proceedings of the 13th international conference on flexible automation & intelligent manufacturing, 9-11 June 2003, Tampa, Florida, USA, pp. 655-667

[WeWe-01] C. Weber, H. Werner, "Objektorientiertes Modellieren am Beispiel der Konstruktion eines Segelbootes", Informationsverarbeitung in der Produktentwicklung 2001 – Effiziente 3D-Produktmodellierungfortschritte und Fallstricke, VDI-Berichte No. 1614, 2001, pp. 95-109

[XuWe-96] S.G. Xu, K.J. Weinmann, "A three-dimensional FE study of sheet metal flow over the drawbead", Society of automotive engineers Inc, 1996

[ZhCE-99] J. Zhang, J. Chen, H. El-Mounayri, "A generic template for collaborative product development", Industrial virtual reality: Manufacturing and design tool for the next millennium, ASME, 1999

[ZiHH-02a] J.U. Zimmermann, S. Haasis, F.J.A.M. van Houten, "Applying universal linking of engineering objects in the automotive industry – Practical aspects, benefits, and prototypes", Proceedings of the Design engineering technical conferences/Design automation conference, 29 September – 2 October 2002, Montreal, Canada

[ZiHH-02b] J.U. Zimmermann, S. Haasis, F.J.A.M. van Houten, "ULEO – Universal Linking of Engineering Objects", CIRP Annals, Vol. 51/1, 2002

[Zimm-05] J.U. Zimmermann, "Informational integration of product development software in the automotive industry – The ULEO approach", Ph.D. Thesis, University of Twente, 2005

[Zura-03] T. Zurawka, "Automotive systems engineering – Lösungen zur Beherrschung der durch die Einführung von Elektronik entstehenden Komplexität", 7. Handelsblatt-Jahrestagung Automobiltechnologien – Vision Automobil, Munich, 9 – 10 April 2003

Abbreviations and Acronyms

CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CAI	Computer-Aided Innovation
CAM	Computer-Aided Manufacturing
CAPP	Computer-Aided Process Planning
CAx	Comprehensive term for all computer-aided technologies
CE	Concurrent Engineering
CSE	Concurrent Simultaneous Engineering
DMU	Digital Mock-Up
DSM	Design Structure Matrix
EO	Engineering Object
FE	Finite Element
FEMEX	Feature Modeling Experts
FROOM	Feature- and Relation-based Object-Oriented Modeling
GCO	Generic Component
IDEF0	Integration Definition for Function Modeling
IT	Information Technology
KAE	Knowledge-Aided Engineering
NC	Numerical Control
OEM	Original Equipment Manufacturer
PDM	Product Data Management
PLM	Product Lifecycle Management
SADT	Structured Analysis and Design Technique
STEP	Standard for Transfer and Exchange of Production Data
ULEO	Universal Linking of Engineering Objects
VPD	Virtual Product Development

2D	2-Dimensional
3D	3-Dimensional

Figure Index

Table Index

ISSN 1860-5990
ISBN 3-86644-033-2

www.uvka.de