Conception Intégrée de Meubles Réalisés en Panneaux de Fibres ou de Particules

K. Pimapunsri

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque
|__|__|__|__|__|__|__|__|__|__|

T H E S E

pour obtenir le grade de

DOCTEUR DE L'INP Grenoble

Spécialité : **« Génie Industriel »**

préparée au laboratoire

Sciences pour la Conception, l'Optimisation et la Production de Grenoble (G-SCOP)

dans le cadre de **l'Ecole Doctorale « Organisation Industrielle et Systèmes de Production »**

présentée et soutenue publiquement

par

Kusol PIMAPUNSRI

le 13 novembre 2007

Conception Intégrée de Meubles Réalisés en Panneaux de Fibres ou de Particules

Directeur de thèse : Serge TICHKIEWITCH

JURY

M. SHPITALNI	Président	Professeur, Israel Institute of Technology, Israël
M. LOMBARD	Rapporteur	Maître de Conférences, Université Henri Poincaré, Nancy
P. MARTIN	Rapporteur	Professeur, ENSAM CER de Metz, France
S. BUTDEE	Examinateur	Professeur, KMITNB, Thaïlande
S. TICHKIEWITCH	Directeur de thèse	Professeur, INP-Grenoble, France

Acknowledgement

First of all, I would like to thank all committee members for their acceptance of being present in my defense and their attention on my work. I would like to thank Professor Moshe Sphitalni who had spent a long time for a journey from Israel and his acceptance of being the president of the committees. I would also like to thank Dr. Muriel Lombard and Professor Patrick Martin who have examined carefully this manuscript and their valuable remarks.

I would like to say special thanks to Dr. Suthep Butdee for his supports since I was in Master's degree at KMITNB, without his supports I could not have this great opportunity to follow my Ph.D. study. And the most special thanks to Professor Serge Tichkiewitch for very warm welcome and his valuable advises since the short period of training course during my Master's degree and the period of my thesis. I would not have been able to complete my thesis successfully without his supports.

I would also like to thank Frank Pourroy for giving advises on RDM6 and Cedric Masclet for his advises before the defense. I am glad that I have become a member of the Integrated Design team of G-SCOP laboratory. I would like to say very special thanks to permanent members and my colleagues – Bruno, Chaiwat, Alexandre, Vincent, Maria, Rosalinda, Lidia, Robert, Nico, Nattawut, Frédéric, and all the others...

The big part of my memory in Grenoble is the times I have spent with my Thai friends – P'Onn, P'Jod, P'Chat, P'Joom, P'Yo, Toh, Pok, Tong, Ohm, Dokkae, Praew, Dao, Kati & all ODOS students, MaMeow, N'Bim, and all the others. I appreciate their supports since the first day I came in France. I would not forget the great times I have spent together.

My parents, my brothers, and my sisters, I know that they are the ones mostly delighted by my becoming a Ph.D. than anyone else. Thank for their supports throughout my entire life.

Table of Contents

List of Figures

List of Tables

Introduction

Can we avoid thinking about the word “globalization” when speaking of the development of design process? What is the real significance of this word to the design process? Why are we concerned by the globalization? One may ask himself by these questions. Furthermore, “concurrent engineering”, “collaboration”, “integration” may well be included in the topic. Let us recall to the goal of a product design in which the fundamental issues that we must take into account are: quality, cost, time, and recycling (QCTR). In the 1970s, the cost of products was the main lever for competitive advantage. Producer desired to reduce the cost of a product (material cost, labor cost, transportation cost, etc.). Many producers had established their factories where the resources were cheaper and easy to be acquired. We may consider that this movement began the globalization. Later in the 1980s, quality superseded cost and became an important issue. Various techniques and methods had been conceived and developed to improve the quality of the product. Later on, intense competition due to the shrinking product life cycle increases customer expectations. Customer does not focus solely on low unit cost and high quality of products but factors such lead time to market, and product customization issues are assuming to be the major role in defining the success of organizations. Today, recycling has been considered as one of the major issues. It concerns both of the environmental issue and the economy issue.

The globalization becomes more predominant since the technology has been broadened, particularly in information technology and communication. The environment of global market and manufacturing has been mentioned regarding to the economy factor and competitive advantages. As a result, many factories have been decentralized toward some countries where low labor cost and material cost, to reduce the product unit cost. Consequently, some sections and/or some of members in team must be distributed in different locations. In addition, to shorten the time from conception to manufacturing, product development phases are required to overlap and executed concurrently. Thus, “concurrent engineering”, “collaboration”, and “integration” have been more mentioned.

Concurrent engineering approach aims to shorten the time to market for product, to develop production process and also to reduce the cost, by performing various engineering activities in parallel as cross-functional team. Nevertheless, due to the increasing of complexity of product design, CE approach outputs numerous of decisions, which may lead the design actors having later some conflicts. In this situation, collaborative engineering approach has become necessary. This approach aims at supporting the individuals in the design team to work together toward a common goal and finding solutions that are satisfying to all concerned. This approach facilitates the design actors by bringing them into a collaborative environment and gives them communication channels for resolving design conflicts. However, it does not assure that they can perform effectively the design activities. In addition, some misunderstandings during meetings may lead to increase the development time and cost. At this time, an integrated design approach is proposed. In the context of integrated design, any actor who intervenes at any time of the product life cycle is required to be presented in the design process in order to share and to exchange their information with the team for developing the product design. Integrated design means to merge different competences of different actors, to solve contradictions between disciplines, and then to integrate knowledge into product design.

Wood furniture industry is one of the highest competitions, which has been impacted from the globalization. It is mostly concerned with short product life cycle and rapid change of models and styles. In this study, we propose an integrated design for wood furniture made of particleboard and medium-density fiberboard. The growth of this sort of furniture has been regularly increasing for a long time. The key factors of this growth are the low price of product, the modern styles and the design as ready-to-assemble (RTA). Although the selling price attracts the customer and increases the demand of products, it also brings companies into a high competitive environment. In order to stay in such environment, the companies need to innovate and to create rapidly new products which satisfy as much as possible the customer requirements.

In the design process, the product design involves various disciplines, each discipline concerns on different objective. As a result, the design team would encounter some difficulties in gathering information, communication, cooperation, and/or making decisions due to the decentralization. This might cause redesign processes and delay entrance to market. Furthermore, companies require being more competitive by putting an effort into the system to satisfy customer’s requirements as much as possible although it may create additional complexity. Therefore, the design process must be developed to satisfy the complex design products.

This study aims at reducing the imaginary complexity in the design process, supporting the design actors to share their knowledge before performing the product design, and exchanging information and constraints during the design process. And finally, it aims to permit the design actors to work together in a virtual collaborative environment. In order to accomplish these aims, we have to integrate these actors to work together as a multidisciplinary design team. We emphasize that the design actors must be able to communicate, to share and to exchange information for solving the design problems and complexity. In this study, a cooperative design modeller (CoDeMo) is proposed. One of the main objectives of CoDeMo is to create a collaborative environment as a virtual meeting room that allows different members, who connected to the network, to participate either in synchronous or asynchronous mode in a design project. With methods and models for integration, the system permits the members to contribute their knowledge into the design project, to access a shared database, to exchange information, to discuss on design problems, to negotiate and to compromise for solving the design complexity. Each member can also employ a tool or a specific application to solve the design problems and/or to evaluate the design.

This thesis consists of three parts. The first part corresponds to the state of the art that includes an introduction to furniture made of particleboard and fiberboard, philosophies of engineering design, and studying of existing engineering approaches. The integrated design approach is developed for manipulating various aspects of product life cycle into the product design and solving problems of complexity in the design process. This part is decomposed in three chapters as following:

- The first chapter introduces the general idea of furniture made of particleboard and fiberboard. It observes the growth of wood furniture market and notices the importance of studying in furniture made of particleboard and fiberboard. It presents then different aspects between massive wood furniture and this sort of furniture, an introduction of particleboard and fiberboard, and a description of different types of such furniture.
- The second chapter examines existing and current approaches of engineering design process. It points out the problematic and some difficulties of the design process, and limitations of the existing approaches. It extracts the pertinent issues of the examined approached to this study for developing an integrated design approach.
- The third chapter aims at understanding the principles of design called Axiomatic Design. It describes extensively the problematic design by introducing the theory of complexity. The four different types of complexity defined by Suh are presented. The inherent complexity in the engineering design process is consequently examined.

The second part introduces concepts toward integrated design. Models and methods for integration, which have been developed in this study and by the integrated design team of G-SCOP laboratory, are presented. These methods and models enable the system a collaborative environment and permit members from different disciplines performing design tasks in a collaborative manner. A method to reduce the time-independent imaginary complexity in the design process is proposed in this part. It presents the interactions between actors during the design process and also presents how the design actors constitute knowledge model and integrate into the product. This part is also decomposed in three chapters as following:

- The fourth chapter examines the previous studies conceived by the integrated design team of G-SCOP laboratory. It aims at presenting models and methods for integration which are the core of integrated design. We apply the concept of product model to store the product data and knowledge of different competences and use the multidisciplinary concept to facilitate the design actors to present their information to the design team. This chapter presents the methods that facilitate the design actors to communicate, to share information, to discuss and to negotiate on the design.
- The fifth chapter aims at presenting the constitution of knowledge model of the design actors. The design team consists of different design actors from different domain of competences. Each design actor is requested to describe characteristics and behaviors of the product. This chapter presents how the design actors contribute such information into the design process. This contribution enhances the design team to share and to exchange their information during the design process.
- The sixth chapter ends this part by proposing a method for reducing the time-independent imaginary complexity in the design process. In this study, we take into account principally three domains of competence: assembly, mechanic, and manufacturing. Therefore the interaction between the design actors in these domains is presented. It also presents how the design actors deal with the design problems in the context of integrated design.

This third part aims at validating the integrated design system and specific applications in applying with products of wood furniture made of particleboard and medium-density fiberboard. This part contains only one chapter.

- The seventh chapter demonstrates the integrated design process. We employ CoDeMo to create a collaborative environment and to bring the design actors into such collaborative environment for working together in a virtual meeting room. It validates the use of features and production rules, and presents interactions between design actors from different trade views during the design process. The system permits the design actors to use their specific applications in order to evaluate product design. This chapter dedicates one section for presenting a specific application, which we have developed, using in wood furniture industry.

The conclusion summarizes the principal results of this study. It also presents perspectives and projections of future work that should be developed.

Part I

State of the Art

Chapter 1

Introduction to Furniture Made of Particleboard and Fiberboard

This chapter introduces the general idea of furniture made of particleboard and fiberboard. On one hand this chapter observes the growth of wood furniture market, and it is to notice the importance of study in furniture made of particleboard and fiberboard on the other hand. This chapter presents the different aspects between massive wood furniture and the furniture made of particleboard and fiberboard. It also contains the introduction of particleboard and fiberboard, and finally the description of different types of such furniture

1.1 Introduction to wood furniture

Wood has been exploited to make furniture since the time of ancient civilization of Egyptians (about 3000 BC) [Thomas 2004]. From the Middle Ages (from the fall of the Roman Empire) through the period of the second World War, the majority of furniture was primarily made of natural massive wood, such as oak, pine, walnut, mahogany, ebony, satinwood, etc. [Pixler 1999]. Wood laminates had been introduced more than three thousand years ago by the Egyptians [Bodig and Jayne 1982]. A thousand years ago, the Chinese shaved wood and glued it together to use in furniture [APA 2005]. However, the furniture made of wood composites, such particleboard, has become prominent since the early 1950s due to the material shortage during the period of the World War II [Wikipedia 2007].

1.1.1 Growth trend of furniture industry

The furniture industry is one of the highest competitions in global manufacturing environment. In 2005, the world's production of furniture is worth about 220 billion euros. It is forecasted that the growth will rise up to 1000 billion euros in 2050 [De Turck 2005]. During 1995 to 2005, the United States, the world's largest furniture importer, increased very largely from 6.5 billion US dollars up to 23.8 billion US dollars [FFE 2006]. More than 60 percent of all imported household furniture is wood. At the "Outlook for the Furniture Markets" seminar organized by CSIL Milano in Italy, the forecast of international trade of furniture was expected to reach 82 billion US dollars in 2005, to 90 billion US dollars in 2006, and 97 billion US dollars in 2007. In the last decade of 2005 to the year 2006, the great demand of furniture's consumption was highly augmented and it eventually exceeded the supply of furniture production

Europe was the largest furniture market of the world. In 2004, Europe market based on 25 countries, the total apparent consumption of furniture was 95.6 billion euros. Compared with the year 2003 value, the consumption's value increased to 95.5 billion euros or 1.1% upward. This production accounted for some 43.1% of the global production [UEA 2005]. Observations of many market researches indicate that growth trend of wood furniture in this decade (2000 to 2010) will be increasing continually. According to this study, the outlook for the industry of furniture made of particleboard and fiberboard is observed. Focus first on the trends and projections for the production of particleboard and fiberboard in Europe. Figure 1.1 shows the outlook for particleboard production in Europe to 2020 under the baseline scenario. Overall, production is expected to increase by the average annual rate of 2.6 percent, with the increase in production from 40 million m^3 in 2000 to 67 million m^3 in 2020 [UNECE 2005].

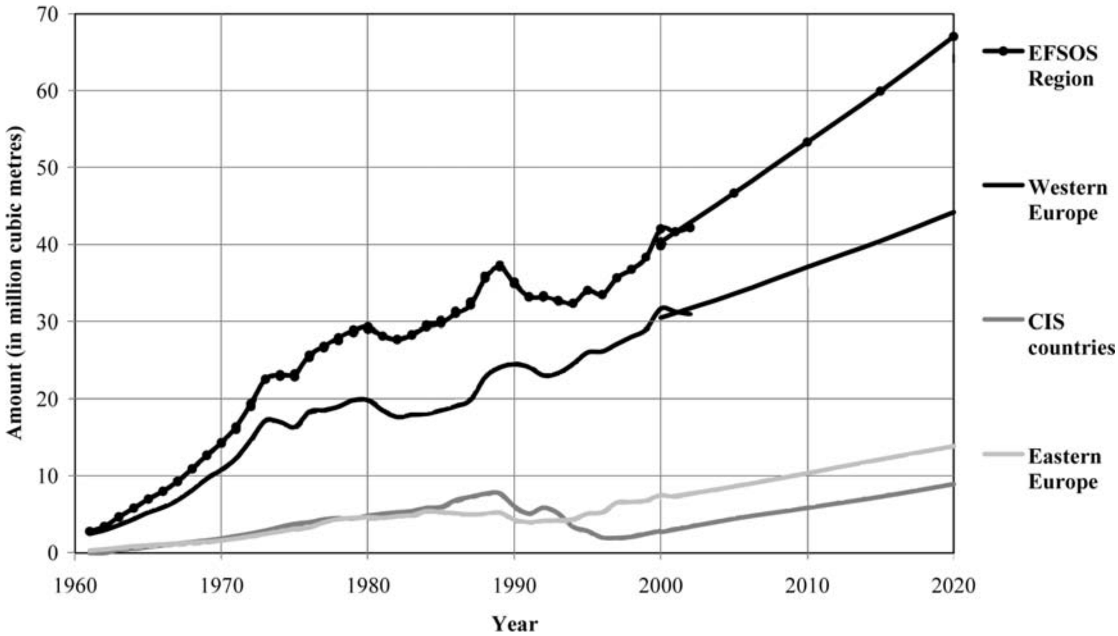


Figure 1.1 Trends and projections for the production of particleboard in Europe [UNECE 2005]

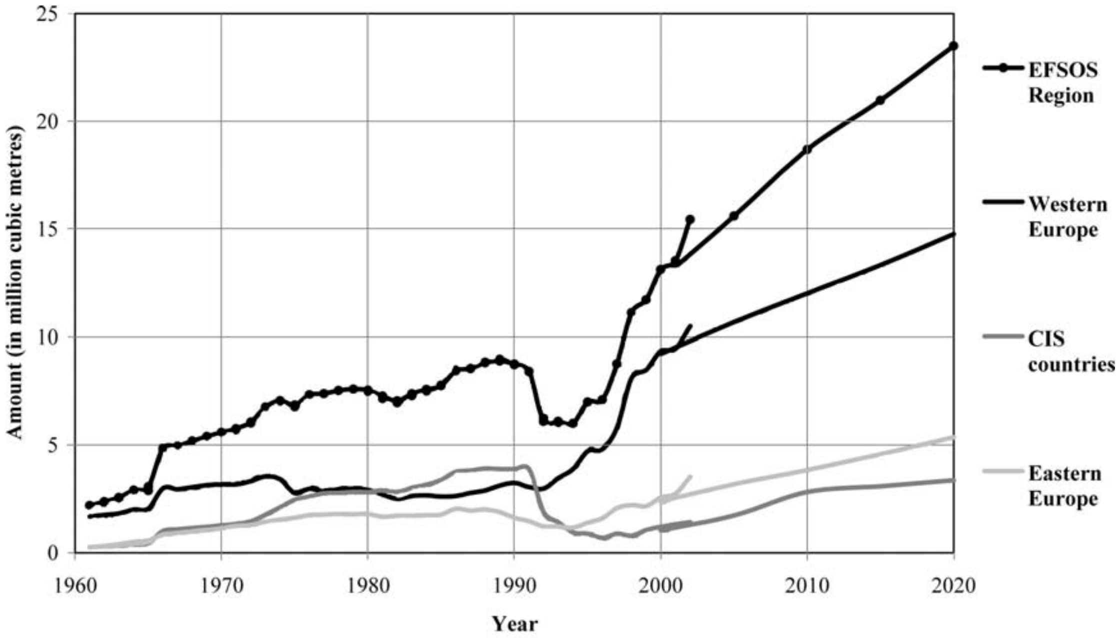


Figure 1.2 Trends and projections for the production of fiberboard in Europe [UNECE 2005]

The outlook for fiberboard production to 2020 is shown in Figure 1.2. A whole production for Europe is expected to increase at the average annual rate of 3.1 percent. Production of fiberboard will be likely doubled over the next 20 years, with production increasing from around 12.7 million m^3 in 2000 to 23.5 million m^3 in 2020 [UNECE 2005].

However, the latest statistics from United Nations Commodity Trade Statistics Database confirm that China became the world's largest exporter of wooden furniture in 2005. From 1995 to 2005, the total value of wooden furniture exports rose seven-fold from 932 million US dollars to7.15 billion US dollars, and wooden furniture exports accounted for only about one-quarter of China's total furniture output in 2005[1].

As a result, this is essential to observe the trends and projections for the production of particleboard and fiberboard in China. Figure 1.3 shows the projected outlook for particleboard production in China to 2010. The study of [Lyons 1997] has predicted that during 2005 to 2010, the production of particleboard had been increasing from 8.92 million m^3 to 12.5 million m^3. Say that the production is expected to increase to 42%. These trends indicate obviously the rapid growth both of production in China during 1995 to 2005.

[1] Source: news headlines from http://www.furnitureglobal.com on November 3, 2006

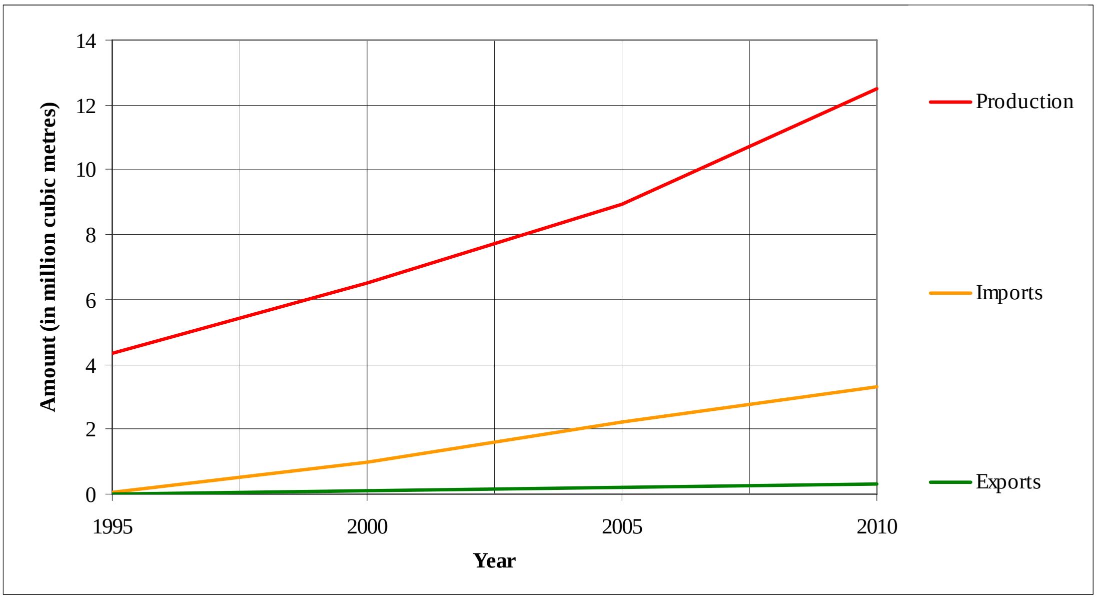


Figure 1.3 Trends and projections for the production of particleboard in China[2]

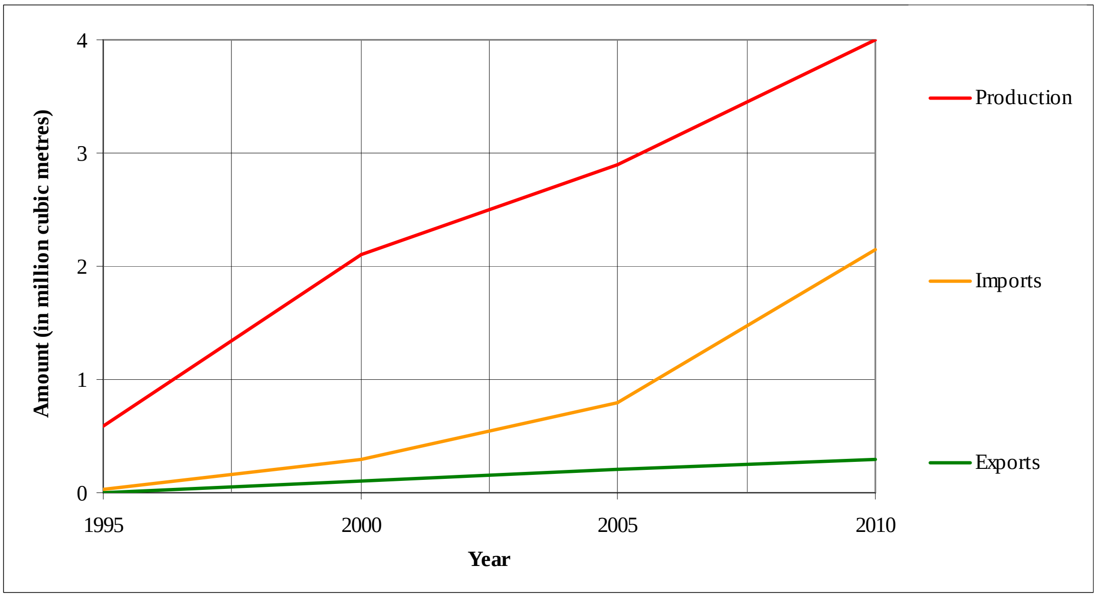


Figure 1.4 Trends and projections for the production of medium-density fiberboard in China[2]

Figure 1.4 shows the projected outlook for medium-density fiberboard production in China to 2010. Lyons has predicted that, during 2005 to 2010, the production for China is expected to increase from 2.9 million m^3 to 4 million m^3.

[2] Source: trends derived from FAO Asia Pacific Outlook Study On Wood Based Panels - 1995 – 2010 from Lyons (1997)

Production of medium-density fiberboard increases almost seven times over the next 15 years.

From these four figures, one can notice the continuous growing trends and projections for the production of particleboard and fiberboard. Furniture is one of the first consumed products of human. Once the census and projections of world population are considered, it is not surprising why the trends and projections of particleboard and fiberboard have been forecasted as shown in the figures. [U.S. Census Bureau 2006] has observed that in 2000, the world population numbered around 6 billion, increasing from 5.2 billion in 1990. It will increase continuously to 6.8 billion in 2010 and to 7.6 billion in 2020 respectively. These projections of world population reflect the growth trends and projections of particleboard and fiberboard that consequently articulate on the importance of the study in furniture made of particleboard and fiberboard.

1.1.2 Benefits of using non-massive wood

Due to the material shortage during the World War II, the notion of using wood composites, such particleboard and fiberboard, as the replacement of massive wood in some applications had been more attention. In the same period, North America has experienced rapid growth in the use of both structural and nonstructural wood composites, substituting primarily for traditional massive wood products [Smith and Wolcott 2006].

Wood composites are made from wood-based materials which can be veneer, strands, particles, fibers, etc. These materials are bonded together with a synthetic adhesive using heat and pressure. The characteristics of wood composites are essentially depending on the nature of the wood raw material and the adhesive. These characteristics include mechanical properties, water resistance, dimensional stability, surface quality and machine ability.

The production of wood composites has increased dramatically over the past three decades due to a number of factors. The changing wood supply, the development of new composite technologies, and the widespread acceptance of architects and builders have each contributed to increased wood composite production [Kirkpatrick and Barnes 2006]. As a result of scarce of logs, the demand of wood composites is forecasted to increase. Wood composites are widely used in various manners, often similarly to massive wood. Moreover, in many applications, wood composites contain themselves the practical priority to massive wood due to certain comparative advantages:

- Customization for applications: Since wood composites are artificial wood. It can be designed to meet the application-specific requirements such as dimension, shape, mechanical properties. In addition, using artificial covering materials, for instance, veneers and edge bands would be used to facilitate the product design and also to create many more attractive aspects of product.

- Procurement: Due to tightening of logging restrictions of natural forests in many countries, it has become increasingly difficult to procure raw wood used in wood manufacturing. But wood composites do not need a large log to construct a large panel.
- Cost: Wood composites are less expensive when manufactured in large scale since they are made of leftover wood scraps. It is typically made from small wood particles such as sawdust, planer shavings, wood residues, etc.

Resource Conservation Alliance [Hayes 2006] also states the advantages of using wood residues, which can be made particleboard and fiberboard, in three-fold: economic, environmental, and technological.

1.1.3 Classification of wood composites

[Bodig and Jayne 1982] have listed the most important wood composites in that period and classified into six groups, i.e. massive wood, modified wood, layered composites, particle composites, fiber composites, and flour composites.

[Youngquist 1999] [3] has classified wood-based composites in the ***Wood Handbook*** as shown in Table 1.1. These classifications were modified from the original version of [Maloney 1986] in order to reflect the latest product developments at that time. However, these traditional composites can be placed into three main groups based on particle size: veneer-based, particle-based, and fiber-based materials as shown in Table 1.2. The details of these groups and sub-groups can be found in [English et al 1994].

This study interests in particle-based material and fiber-based material. Particleboard and medium-density fiberboard are mainly used in furniture industry in particular.

[3] Youngquist has classified wood composites in rev.1999 that is the same as rev.1987 in the Wood Handbook

Table 1.1 Classification of wood-based composites [Youngquist 1999]

Veneer-based material
- Plywood
- Laminated veneer lumber (LVL)
- Parallel-laminated veneer (PLV)

Laminates
- Laminated beams
- Overlay materials
- Wood-nonwood composites

Composite material
- Cellulosic fiberboard
- Hardboard
- Particleboard
- Waferboard
- Flakeboard
- Oriented strandboard (OSB)
- COM-PLY

Edge-adhesive-bonded material
- Lumber panels

Components
- I-beams
- T-beams panels
- Stress-skin panels

Wood-nonwood composites
- Wood fiber-plastic composites
- Inorganic-bonded composites
- Wood fiber-agricultural fiber composites

Table 1.2 Types of commercial lignocellulosic composites [English et al 1994]

Veneer-based material
- Plywood
- Laminated veneer lumber (LVL)

Particle-based material
- Waferboard and Oriented strandboard (OSB)
- Particleboard
- Cementboard

Fiber-based material
- Insulation board
- Medium-density fiberboard
- Hardboard

1.2 Furniture made of particleboard and fiberboard

Particleboard was originally developed in Europe and was first produced industrially in the late 1940s in Germany [Canadian Forest Industries 2006]. It was introduced into the United States in the early 1950s [Youngquist 1996] and was produced industrially since 1960s. However, fiberboard has been manufactured since 1914 to use in broad spectrum of housing and building applications [Brenden and Schaffer 1980]. Firstly, particleboard intended to be a replacement of natural wood. In the early 1950s, particleboard started to come into use in furniture but, in many cases, it remained more expensive than massive wood. Late after, the technology of particleboard manufacturing was highly developed, particleboard became cheaper and better in quality. This evolution created rapidly the change in number of particleboard and fiberboard industry.

1.2.1 Definition of particleboard and fiberboard

There are numerous definitions describing “particleboard”. For having a same coincidence, this brings the verbatim term of particleboard from American Society for Testing and Materials [ASTM 2005] defined as follows:

“A generic term for a panel manufactured from lignocellulosic materials (usually wood) primarily in the form of discrete pieces or particles, as distinguished from fibers, combined with a synthetic resin or other suitable binder and bonded together under heat and pressure in a hot press by a process in which the entire interparticle bond is created by the added binder, and to which other materials may have been added during manufacture to improve certain properties. Particleboards are further defined by the method of pressing. When the pressure is applied in the direction perpendicular to the faces as in a conventional multiplaten hot press, they are defined flat-platen pressed and when the applied pressure is parallel to the faces, they are defined as extruded.”

The particleboard industry grew up by the need to dispose the large quantity of sawdust, planer shavings, the use of mill residues and other relatively homogeneous waste materials produced by other wood industries. Particleboard is now widely used in the manufacture of furniture such as cabinets, floor underlayment, shelving and many other products.

The term “fiberboard” includes hardboard, medium-density fiberboard (MDF), and insulation board. The difference between particleboard and fiberboard is that fiberboard exploits the inherent strength of wood to a greater extent by grinding up wood materials into small pieces like fiber-like material and recombining these fibers with adhesive intertwining of the fibers being the primary binding agent forming the board. This makes fiberboard is denser and stronger than particleboard.

Fiberboard, particularly MDF, is frequently used in place of massive wood, plywood, and particleboard in many furniture applications. Compared to particleboard, MDF has a very smooth surface, which facilitates wood-grain printing, overlaying with sheet materials, and veneering [English et al 1994]. MDF is widely used in the manufacture of furniture such as cabinets, door parts, millwork and laminate flooring.

Figure1.5 Examples of particleboard and fiberboard

1.2.2 Different aspects between massive wood furniture and furniture made of particleboard and fiberboard

At the beginning, furniture made of particleboard and fiberboard seemed inferior in quality. People had a low impression of this sort of furniture as a result of its weakness characteristics. Conversely, furniture made of massive wood was usually the first choice of the customer. Massive wood can be sculptured for being high privileged furniture. In addition, it is stronger and more durable. This is a reason why furniture made of massive wood has been used until the present day. However, the wood furniture industry has been changed due to a number of factors: rapid changes of the innovation and technology, difficulty of procurement of natural wood, and environment aspect.

Particleboard and fiberboard take now the pivotal influence on furniture industry. In comparison with massive wood, particleboard and fiberboard still lack durable aspect and strength to resist a large weight as massive wood does. Nevertheless particleboard and fiberboard are placed in priority to massive wood due to certain comparative advantages:

- Cost: The most important factor that influences both of customer and producer to choose particleboard and fiberboard is the selling price for customer and the cost of material for producer.
- Stability: Massive wood is likely to be warped and split by humidity whereas particleboard is not. This stability enables new design possibilities, without any considerations pertaining to the seasonal variation.

- Attraction: Many people consider that massive wood furniture is more attractive than particleboard and medium-density fiberboard. However, as a result of the adaptation of veneer appearances, particleboard and fiberboard have claimed their place on attractiveness from the customers' eyes. Furthermore, various edge bands are also used for banding the edges of furniture surface which will be visible. In Figure1.6 shows example of veneers and edge bands. These edge bands can be made from PVC, ABS material or melamine.


Figure1.6 Examples of veneers and edge bands

One could notice the major disadvantage of furniture made of particleboard and medium-density fiberboard that it is very prone to expansion and discoloration due to moisture. However, the advantage of these veneers and edge bands is not only making furniture to be attractive but also keeping furniture a resistant due to moisture. Some parts of this sort of furniture are now made-up for using in some places of bathroom, kitchen and laundry.

1.2.3 Classifications of furniture made of particleboard and fiberboard

We may classify wood furniture into three categories as following:

- Wood furniture made of massive wood, as represented in Figure 1.7.
- Wood furniture made of wood composites such as particleboard, fiberboard, medium-density fiberboard, oriented strand board, etc., as represented in Figure 1.8.
- Wood furniture made of both massive wood and wood composites, as represented in Figure 1.9.

Figure1.7 Examples of wood furniture made of massive wood

Figure1.8 Examples of wood furniture made of wood composites

Figure1.9 Examples of wood furniture made of massive wood and wood composites [IKEA[4]]

4 From the site www.ikea.com

In this study, we focus only on the wood furniture made of particleboard and medium-density fiberboard, which is "ready-to-assemble" (RTA) or is called "knock-down" furniture.

Historically, furniture is manufactured and assembled at the factory and then delivered to the distributor or customer as a complete unit. Sometimes the product is too bulky to deliver to the customer from one location to another. Additionally, the size of the furniture may be impossible to be delivered to certain destinations because of dimensional constraints in hallways, doorways and stairways. As a result, customers may have limited selection of furniture because of the size and weight of fully assembled furniture. To overcome such shortcomings and to provide several options in the storage, delivery and transporting of furniture, IKEA, a well-known Swedish furniture company, developed a new sort of furniture which is widely known as self-assembly design. This sort of furniture is apparently a mass production in an effort to gain a low price, to reduce the cost of production and transportation by using a flat-pack distribution method.

Knock-down furniture, also known as "ready-to-assemble (RTA) furniture" or "flat packs", is designed for self-assembly. It is supplied as a kit of flat parts and fasteners to be assembled, usually by the end user, with simple tools. IKEA is a pioneer in self-assembly design. Products of this sort of furniture are usually a single unit. This sort of furniture is apparently a mass production in an effort to gain a low price. Figure 1.10 shows by example a desk which made of particleboard and fiberboard. As a result of the self-assembly design, it does not need any special skill in assembly.

Figure 1.10 Example of knock-down furniture [IKEA[4]]

With diversity of fitting hardware, customers can assemble products by themselves. This consequently permits producers to reduce cost of assembly. Furthermore, the flat-pack distribution method by using of packaging also reduces cost of transportation by not shipping air. These advantages allow the producers to introduce a better price to the market and are the key factors of the rapid growth of this sort of furniture.

Figure 1.11 Examples of fitting hardware

1.3 Summary

In this chapter, it has noticed the importance of the study in furniture made of particleboard and fiberboard. The growth trend of this sort of furniture has been illustrated. The selling price of product is the prominent factor that drives the industry of this sort of furniture grows rapidly. However, this factor also brings companies into a higher competitive environment. In order to stand in such environment, the companies have to improve themselves in these principles: cost, time, and quality. Moreover, the product design must satisfy the customer's requirements as much as possible. In the following chapters, the existing approaches and principles that can be applied to develop the industry of this sort of furniture are presented.

Chapter 2

Engineering Design Process

It is well known that the design process is the crucial activity in the product life cycle. This is why a number of evolutionary changes in the area of design have been endeavored since the past couple of decades. Our study concerns the development of methods and tools that allow design actors [5] to work in collaboration and integration. However, before such a system can be proposed, the understanding of how the design actors develop the design process and which methodology they need to perform design, are required. This chapter starts first with the introduction to fundamentals of engineering design process that are widely used. Extracting the pertinent issues to this study, the chapter examines the existing and the current approaches of engineering design process and product design development.

2.1 Introduction

The design process is one of the most critical factors in the product development. Several philosophers have provided the formal description of design process in term of a prescription model and a description model. [Willemse 1997] summarized the major of distinction between prescriptive model and descriptive design model. The goal of descriptive models is to describe and categorize the activities of a designer, in order to understand the functional mechanisms that drive the designer but it does not support the designer in carrying out his task. While the prescriptive models is to provide a systematic description of the activities that a designer should perform in order to fulfill the design task, the prescriptive methods separate the design process in a sequence of phases to be completed. [Erens 1996] has given some examples of these

[5] A design actor means a participant that could be a designer, an expert, an engineer, or a contributor who participates in the design process.

two models in his thesis. By the way, it is not surprising that there have been variations in these descriptions both in terminology and in detail. However, they generally agree in the principle that the design evolves progressively in a step-by-step manner from statement to statement.

The term "design process" can be read as "problem-solving process" which begins with the identification and analysis of a problem or need. It proceeds structurally through a sequence in which information is researched, explored and evaluated until the optimum solution to the problem or need is devised [UK Technology Education Centre 1996]. Yet, design was not a total process. Each participant investigates and imposes unilaterally his/her ideas without taking into account the other's constraints. Therefore the designer was expected to balance all the considerations that came to bear upon the design of particular artifacts, systems and environments. Thus, design is not an activity only for engineers and designers but it is a shared activity among those who design artifacts, systems and environments, those who make them and those who use them.

2.2 Sequential engineering approach

The term, 'sequential engineering', also known in other terminologies such as traditional engineering, conventional engineering, etc., had been in use for decades. Sequential engineering is an approach in which specialists work in a compartmentalized manner. It is characterized by each discipline performs its own individual function and passes the results to the next discipline in the serial chain. Then this section examines design process models which are considered as sequential engineering.

2.2.1 Pahl and Beitz's design process model

The first model presented here is developed by [Pahl and Beitz 1996]. This is one of the most established models of the design process. In this model, the design process consists of four main phases which proceed sequentially. These phases are planning and clarifying the task, conceptual design, embodiment design, and detail design (see Figure 2.1).

- Planning and clarifying the task – are about how to obtain a product idea. It is engaged to the macro view by taking into account current market situation, company and economy. This involves with collecting information concerning to the customer's requirements. These requirements can be either or both of internal requirements from the develop team and external requirements from the consumers. The result of this first step is a detailed product proposal which identifies the list of requirements or design specification.

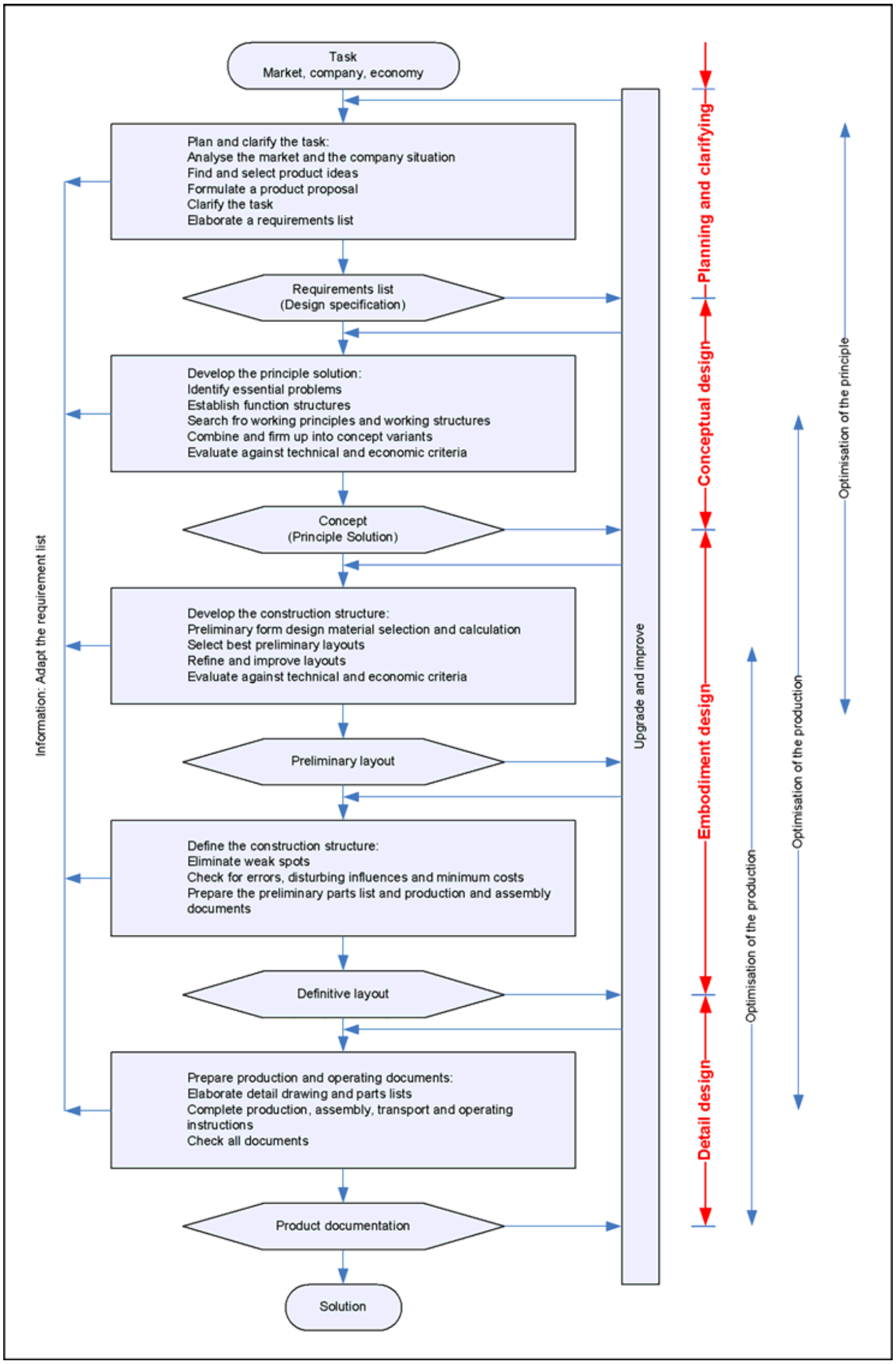


Figure 2.1 Pahl and Beitz's design process model [Pahl and Beitz 1996]

- Conceptual design – is about defining the essential problems and establishing the function structures. In this phase, the problems are decomposed into sub-problems in order to create design concepts followed by the required functionalities. These alternatives design concepts will be evaluated against the specification of physical principles to obtain the most appropriate concept. The result of this step is a design concept which determines the appropriate principles and the working structure. Those principles should be able to satisfy the list of requirements in the first step.

- Embodiment design – is about developing in more details the proposed design concept with those principles and multidisciplinary. This phase typically takes a large proportion of time in the design process. Having a large number of information and corrective steps is possible. This phase requires a great deal of communication and negotiation among designers to meet the specification. This embodiment process is considered to be complex; the simultaneous approach and higher level of information are required. An evaluation against the technical and economical criteria is needed to optimize the design and to evaluate different variants. The result of this step is a definitive layout which is a technical description of entire design such as a drawing, etc.
- Detail design – is about finalizing the definitive layout. This phase has a major influence on production cost and quality. It typically involves finalizing tolerances, dimensions, materials and the detailed manufacturing information. The result of this final step is final production documents of detailed components drawings, of assembly drawings, and of the parts.

2.2.2 Pugh’s design process model

Pugh describes the model of the design process, as shown in Figure 2.2, as the “design core” of the product development process. The major difference between this model and Pahl and Beitz’s model is that Pugh includes the stage of manufacturing and sale into the design process. In addition, an indication is the degree of iteration and feedback from stage to stage. Pugh takes into account a stage approach to the over all process comprising market analysis, through specification, conceptual design, detail design, manufacture, and selling.

From these two models, we can found that the flow of information is mostly one way and is considered severely restricted. It is obvious that each phase has to make decisions and completes tasks before passing information to the following phase without knowing their limitations. Yet, that information does not include any detail or data during the decision making process, but only the results of the process. Each phase is independent and does not take account constraints of other’s phases due to lack of cooperation in decision process. As a result of having insufficient information, it causes consequently the problem of “*over-the-wall*” syndrome [Salomone 1995]. This causes numerous iterative interactions and evaluation processes in and between each phase. Furthermore, it might lead to problems later in the process. The design may be not optimized for the manufacturing process and other aspects. Some changes or redesign processes usually increase cost and time to develop the product, resulting in delay of introducing the product into the market. For example, a design of an automobile consists of thousands parts which have to be assembled together; none of these parts are designed and developed in isolation from each other. The design involves millions of decisions over its life cycle from various engineers and experts.

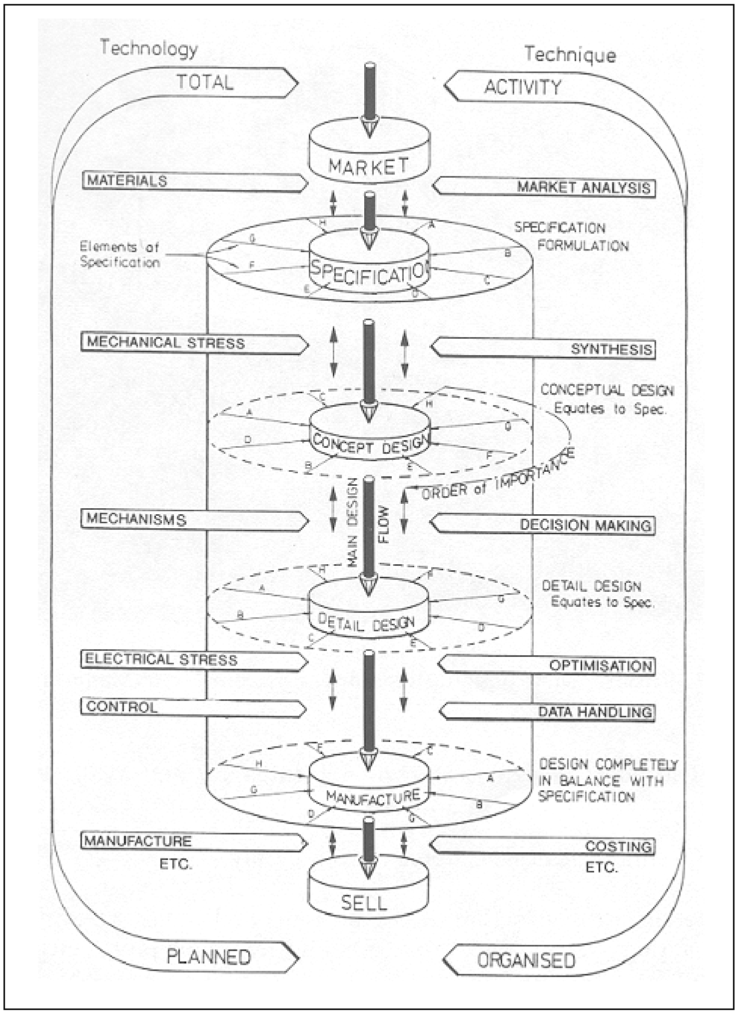


Figure 2.2 Pugh's iterative design process model [Pugh 1991]

To reduce a large number of repetitive activities, this requires a very precise description of result from the *right man* wiht the *right answer* at the *right time*. However, the nature of this iterative design process does not support such requirement. This is why the sequential process was ignored. In addition, [Clausing 1994] formulated 10 cash drains that summarize the problems caused by traditional product development. [Salomone 1995] also has given three primary reasons that caused the design process to evolve into a concurrent process i.e. rapid pace of technology, forced design cycle compression, and emerging information technology and methodologies. Examples of the emerging information technology are: computer-aided design (CAD), computer-aided engineering (CAE), electronic communication (e-mail, e-messenger, etc.) whereas methodologies are such as quality function deployment (QFD), design for X (X can be assembly, manufacturing, etc.), etc.

Since the traditional approach cannot ensure that the design can be processed correctly with minimum time-consuming and low cost. A new approach was asked to merge different functions/phases in the design process using the emerging information technology and methodologies.

2.3 Concurrent engineering approach

Concurrent engineering (CE) was used at the first time in the US in 1989 [Sohlenius 1992]. CE also is referred to as simultaneous engineering, life-cycle engineering, parallel engineering, multi-disciplinary team approach, or Integrated Product and Process Development (IPPD) [Prasad 1996]. CE has become a well-known term since the growing demands for variety of products; customization, high quality and lower cost have made engineering design a very complex activity. The decade after the concept of CE was introduced, there were numerous textbooks and articles have been published about this approach to clarify its definition and conception. Sohlenius defined the meaning of CE as:

"A way of work where the various engineering activities in the product and production development process are integrated and performed as much as possible in parallel rather than in sequence."

[Parkinson et al 2000] wrote that early definitions concentrated on the simultaneous development of product and processes such as above. Other definitions concentrate on the communication in terms of those between various functions inside or even outside the design team. Parkinson and [Kara et al 1999] summarize that the considerations of all downstream activities which are likely to affect the product's life cycle at the products design stage. In respect of product design, the designer or design team should be aware at all stages of implications which the decisions taken at this stage have upon the final manufacturing specification and its resulting outcome.

The difference that distinguishes CE from sequential engineering is that one stage, in the sequential approach, cannot perform tasks without result of the previous stage but CE allows different stages perform possible tasks as soon as possible without waiting for the result of previous stage. Consider the production of an automobile as an example, there are numerous parts must be assembled together to be a car. However, the sprayer can spray the car; the assembler can assemble seats to the car; the wheels can be attached to the transmission without waiting for the engine. As well as in the design process, all phases (before manufacturing) can be incorporated and perform the tasks in parallel as shown in Figure 2.3. However, the disciplines that stay outside the design phase (such manufacturing, selling, service, etc.) must be included during the design process as well. This approach will overcome the problem of "over-the-wall" syndrome. That is to say, the design description has to be completed before passing to manufacturing process.

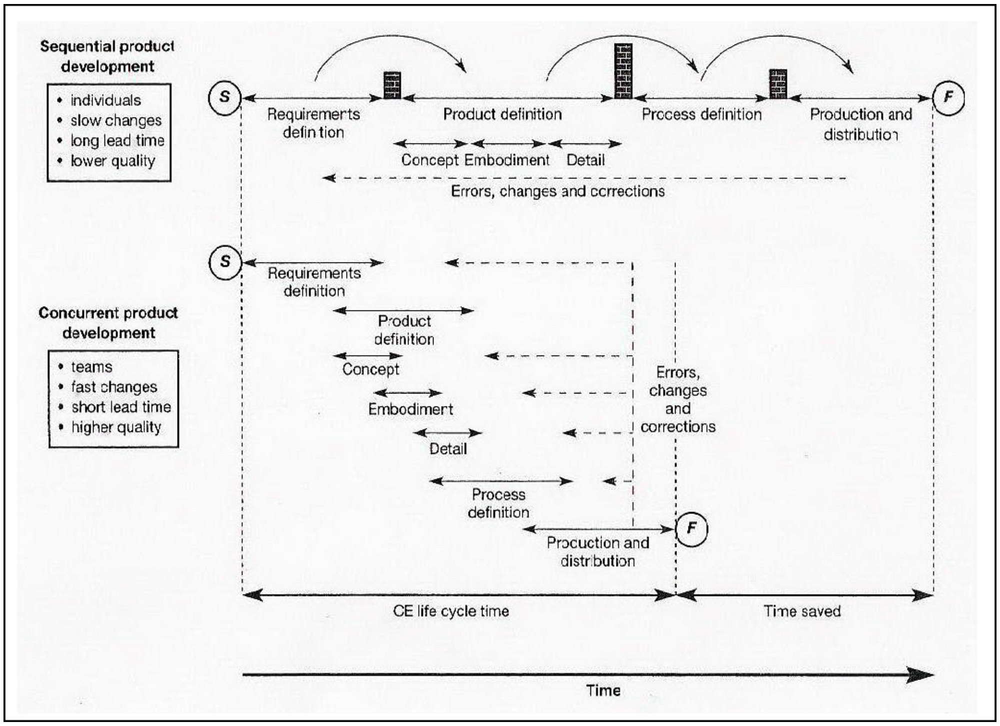


Figure 2.3 Sequential and concurrent product development [McMahon and Browne 1998]

As presented in the previous section, the design process is the role pivot that signifies to the lead time and the total cost of product. Besides, it causes a direct impact on product quality, manufacturing process and cost through the disposal of the product. [Salomone 1995], [Prasad 1996], [Singh 1996] have indicated that most of product's cost is engaged at the early phase of its life cycle. The well known curve in Figure 2.4 presents the cost incurred and committed at different stages in an automobile industry. It indicates that around 80% of total product cost is committed in the design stage. This implies that taking into account various aspects of product life cycle and making decisions at early stages of product development will reach the lower total cost.

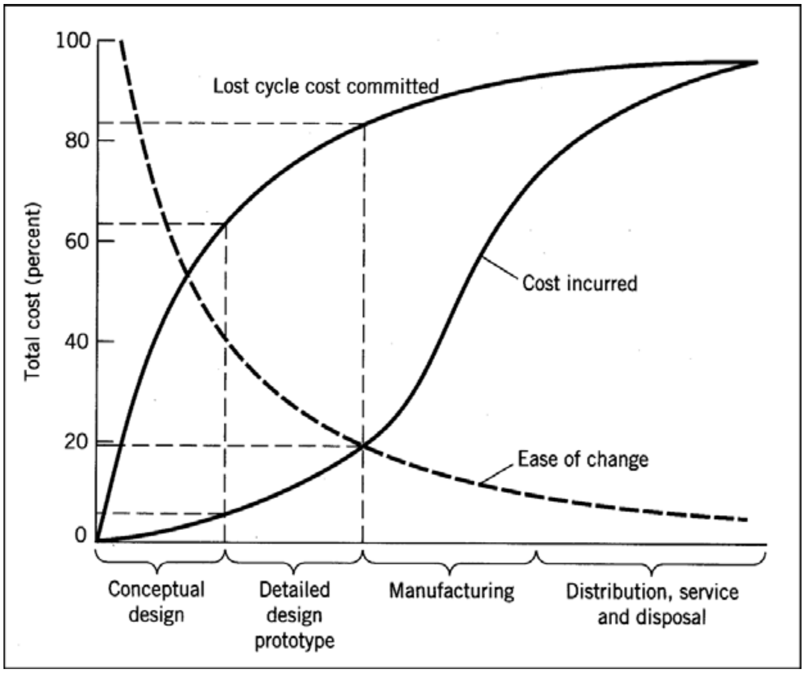


Figure 2.4 Characteristic curve representing cost incurred and committed during product life cycle

The main goal of CE is to shorten the lead time of product development. However, CE also improves quality of product and production process. [Singh 1996] briefly summarized some reported benefits of CE by some of leading companies that not only reduce the develop time but also the lead time, the cost of production, and improve the quality of both product and process.

In the concept of CE, the development of product is regularly relative to manufacturing process and also other support functions. CE approach is considered to make decisions as early as possible and should be done concurrently. Therefore, a basic framework for information flow is provided to collect necessary information from contributors of those various phases and to facilitate the decision making process. Ideally, all relevant information and knowledge should be brought together before making decisions. However, acquiring such relevant and up-to-date information and knowledge across different functions in a large company is complex and time-consuming process [Huang 2004]. [Myint 1999] remarked the practice of CE could not fully achieve the parallel structuring of all activities due to design nature. [Singh 1996] also observed that there are many sources of difficulty in implementing CE, i.e. characteristics of the design process, volume and variety of life-cycle knowledge, and separation of life-cycle functions. This can be summarized that the design process involves a number of activities separated into stages which are further divided into sub-problems. These sub-problems come from different disciplines which are responsible for different functions. This separates designers to concentrate on narrow issues and would ignore the overall problem. Furthermore, a large volume and variety of knowledge from various functions also would lead the designers to concentrate on the optimization of single life cycle factors rather than taking a holistic approach.

Let us consider by example, the production of the Airbus family of aircraft that takes place at various sites across Europe. More than 30 members of the European aeronautics industry based in twelve countries are working together in the construction of major airframe structural components, including composite lateral and center wing boxes and fuselage, metal composite joints and advanced metallic fuselage sections [Pritchard 2002]. Each completed section of the aircraft, such as wings, tail, nose, fuselage of A380, is produced in different country and finally the plane is assembled in France. With this global manufacturing, only collecting all relevant information from contributors and overlapping the activities are not adequate. In addition, the output of CE approach usually is a number of decisions. This would lead the designers having later some conflicts between those functions. A new approach that takes account of discussion, negotiation and compromising is considered necessary. The next section presents such that approach which is called "collaborative engineering" for solving such problem.

2.4 Collaborative engineering approach

Due to the globalization of market and manufacturing, many companies have established their factories in the countries where the resources are cheaper so they could reduce the cost of production (such as labor cost and material cost). Figure 2.5 shows by example a global environment of furniture made of particleboard and fiberboard. Head office is in France, designers work in Sweden, the factory, in which manufacturer, mechanician, and assembler are working, is located in China or Thailand and the customer throughout North America. Under such global condition, designers, experts, and members in team have been decentralized and distributed in different locations around the world. As a result, the design team may encounter the difficulties in gathering and exchanging of information/knowledge and also communication obscurity during the design process. This would lead the designer to make wrong decisions that cause a redesign process and increase the lead time to market of the product. A collaborative design is then needed to solve the distant problem and to bring together the expertise of various designs and engineering disciplines into the design process.

Furthermore, increasing of product complexity, CE approach outputs a number of decisions. Only "multiple decisions" cannot deal with this crisis without a time-consuming process. *Arrow's impossibility theorem* states that, in general, the preference indicated by individuals in a group does not have any value in determining the preference of the group as a whole; see an example in [Suh 2001]. At this time, designers and contributors are being asked to be more mutually considerate as a *social teamwork*. Whether the design process is sequential or concurrent, must involve multiple stakeholders to deal with different expertise and competing interests. With this challenge, designers necessitate a new method of not only multiple decisions but also negotiation and compromising to obtain a single agreement [Lu 2004].

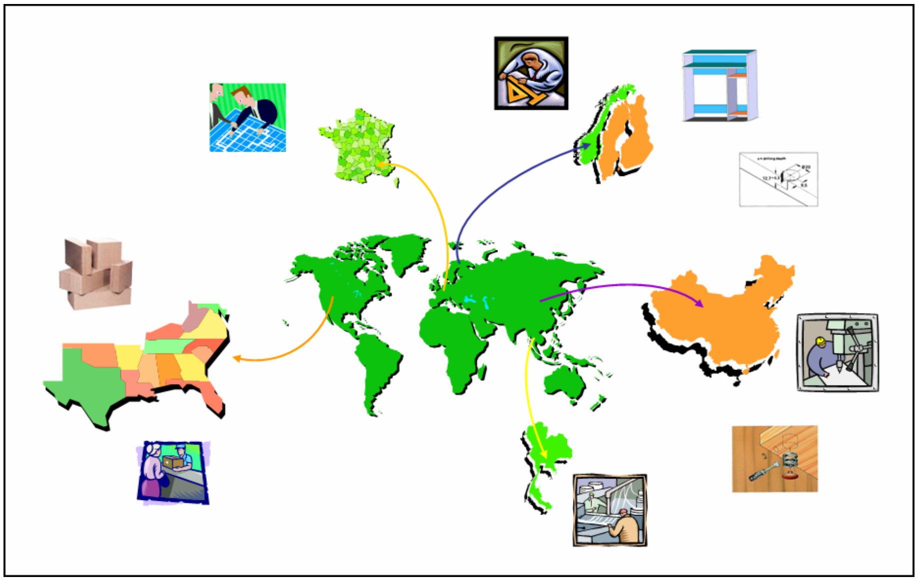


Figure 2.5 Globalization of wood furniture made of particleboard and fiberboard

Before addressing to details of the collaborative approach, it is important to clarify the definitions and differences between coordination, cooperation, and collaboration to properly comprehend the issue.

Coordination is characterized by informal relationships that exist without a commonly defined mission, structure or effort. Information is shared as needed and authority is retained by each organization. [Lu 2007] shows an example of a coordination activity – driving on a busy highway during rush hour. Drivers are in different cars, with different destinations and interests. Each one must carefully coordinate with each other to avoid accidents that would delay traffic and harm everyone's interest. Coordination is defined as the process of managing "uni-directional" task dependencies between activities "across multiple levels" of the hierarchy.

Cooperation is characterized by more formal relationships and understanding of compatible missions. Some planning and division of roles are required, and communication channels are established [Kvan 2000]. The need of being more effectively competitive motivates individuals to work in group with shared resources and methods, as seen by example activities of ants. Cooperation is defined as the process of managing "bi-directional" task dependencies between activities "within the same level" of the hierarchy.

Collaboration is characterized by more durable and pervasive relationships. Collaboration means working together toward a common goal which the team attempts to find solutions that are satisfying to all concerned. [Lu 2007] wrote that collaboration requires a team of individuals to work on tasks that not only have shared resources (as in coordination) and shared outcomes (as in cooperation) but, most importantly, shared common goals. Lu consequently defines the definition of "collaborative engineering" as:

> *"A discipline that facilitates the communal establishment of technical agreements among a team of engineers, who must work together toward a common goal, with limited resources or conflicting interests."*

[Sky and Buchal 1999] presents two types of collaboration: *mutual* and *exclusive*. In the mutual collaboration, designers work together entire the session but only a few semantics are documented. In the exclusive collaboration, the designers work on separate parts but collaborating periodically to inform and negotiate and having more design semantics. The participants in this type of collaboration produced more design semantics than when working individually. To reach such new collaborative approach, the exclusive collaboration is considered necessary.

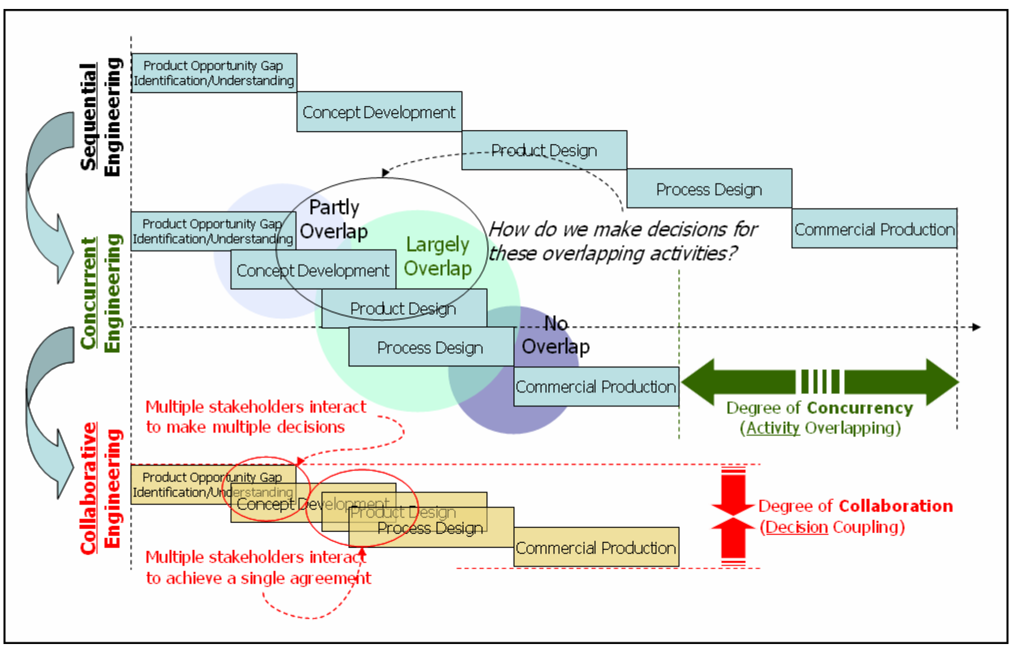


Figure 2.6 Development of engineering design process [Lu 2006a]

To distinguish the difference between the collaborative engineering and the two precedent approaches, [Lu 2006a] has summarized the development of engineering design approach as shown in Figure 2.6. Sequential engineering approach has been replaced with CE approach due to the growing demands for having better quality, lower cost, and faster design. CE approach solves the problem of "over-the-wall" by performing tasks of different phases in parallel as cross-functional team. It achieves moderately in decreasing the lead time by increasing the degree of concurrency in design phases. However, since the product complexity has been increasing, a number of decisions have been increasing as well. In this situation, a collaborative approach has become necessary. The collaborative engineering goes beyond the CE approach by decreasing space between design phases, increasing the degree of collaboration among individuals and teams, and including perspectives of negotiation and compromising for having a single agreement.

Several recent researches have attempted to develop the design process in the collaborative manner. In [Lu 2006b], Lu develops a new approach by proposing a Socio-Technical Framework (STF) for collaborative engineering design. Lu employs a basic questioning method, 3W1H i.e. Who (who are the designer of this decision?), What (what do you want to achieve?), Why (what are the stakeholder rationales?), and How (how do you propose to achieve it?) that are essential for collaborative engineering. The concept is to bind them with spatial relations into two axes: [What → How] called "technical design decisions" and [Who → Why] called "social interaction of design team". Figure 2.7 (a) shows the architecture of the socio-technical framework forming a series of iterative decision making by using the four parameters that associated relations as a "Who → What → Why → How" mapping process for a collaborative engineering. [Who → What] represents the social interaction among participants, [What → Why] constructs a common understanding of task work, [Why → How] establishes a consistent group preference. The next stage is where a joint decision (team agreement) is systematically negotiated by all participants in the collaborative design team.

The new procedure of collaborative engineering design can be expressed in four stages as shown in Figure 2.7 (b). The initial stage interaction is to manage and guide the social interactions, establish the team goal and clarify resources and constraints; understanding is to calibrate, eliminate, or minimize the diverse understanding of stakeholders as much as possible to obtain a common understanding; preference is to rate and to capture the relative strengths of individuals' preference to establish the group preference; decision is to compare and negotiate their preferences for making joint decisions that lead to a robust team agreement.

Many researches attempt to invent a collaborative system that allows designers, experts, and participants to communicate to each other, to share information, to discuss problems, to negotiate and compromise conflicts. However, the design process involves various disciplines: marketing, technologist, assembly, mechanics, manufacturing, maintenance, recycling, etc. Each discipline concerns on different objective. Bringing them into a collaborative environment and facilitate them communication channels for doing meetings does not assure that they can collaborate and perform effectively the design activities. [Sky and Buchal 1999] identify that meetings are the main method of resolving inconsistencies and design conflicts; thus, when misunderstandings occur during meetings, they can lead to increases in development time and design costs. The design system should support the designers to integrate knowledge from different disciplines. As a result, an integrated design approach is proposed in the next section.

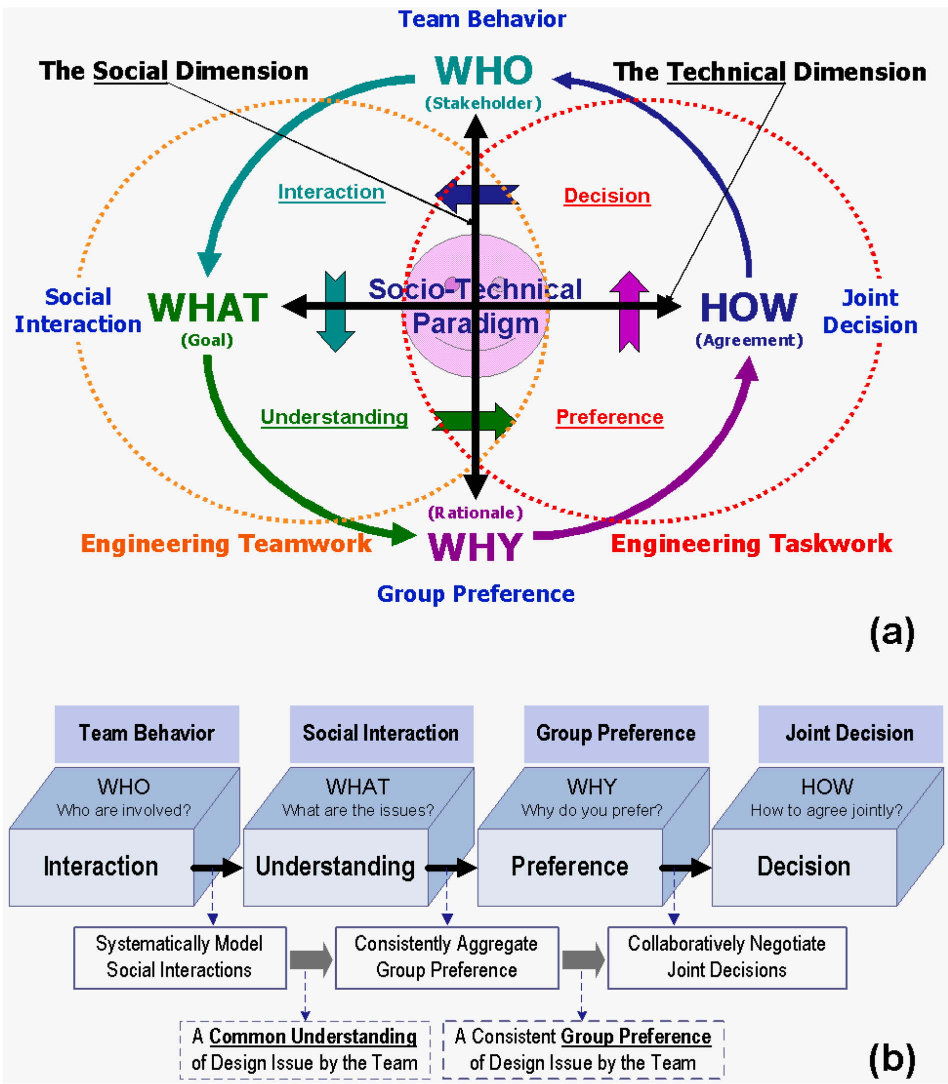


Figure 2.7 A socio-technical foundation for collaborative engineering design [Lu 2006b]

2.5 Integrated design approach

This study aims to develop an integrated design process by taking the benefits of collaborative engineering approach. One can say that integrated design approach seems to concurrent engineering and/or collaborative engineering that is enhanced to develop the design process. After the presentation of consequent problems of traditional designs, in [Tichkiewitch 1990], Tichkiewitch introduced a new step of design that includes models in different domains, an intelligent product database, and database engines, which takes account a multitude of various experts. Consequently, [Tichkiewitch 1994] presented the design process development from a period of CAD/CAM to an integrated design approach. This approach takes into account the manufacturing phase during the design process in order to optimize the final product. One main objective of the integrated design is to reduce the design iterations by taking into account constraints from different disciplines as soon as possible before making a decision. It means that contradictory constraints can be identified and solved earlier than in a non-integrated design approach [Roucoules et al 2003]. [EERE 2005] defines the integrated design as a process of design in which multiple disciplines and seemingly unrelated aspects of design are integrated in a manner that permits synergistic benefits to be realized.

To perform the integrated design, it is not only bringing designers, experts, and contributors into a collaborative environment but the design system has to provide methods and models for integrating knowledge from different disciplines regard to this definition. [Molina et al 2005] concludes that the integrated environment must enforce four dimensions of engineering: *process*, *information*, *organizational*, and *technology*. In addition, we have to integrate data from the whole product life cycle into the design system. PLM[6] systems have been developed to manage collaborative access to product data and to share documents during the design process, cf. [Windchill], [SmarTeam]. Respect to the development of present PLM systems, some limits of still remain. Therefore, IPPOP[7] project has been introduced [Noël 2007], [Noël et al 2004], [Roucoules et al 2006], and [Gzara Yesilbas et al 2006]. The main goal is to provide a data model that can be reached by using external computer services (such as expert application, PLM systems, etc.) related to Product-Process-Organization modelling. The objective of IPPOP project is to manage the design activities, projects, objectives, and resources by integrating three domains of *product modeling*, *design process*, and *industrial organization modeling*.

---

[6] Product Life Cycle Management

[7] Integration of Product-Process-Organisation for engineering Performance improvement, cf. http://ippop.laps.u-bordeaux1.fr, http://projects.opencascade.org/IPPOP

[Sohlenius 1992] defines the three most critical factors: complexity, quality, and lead-time – that determine the competition of a product development as shown in Figure 2.8. He also says that to stay competitive, a product must successfully integrate multiple functions to deal with and to minimize complexity and still meet functional requirements. Due to the increasing of complexity of engineering problems and intense competition in the world market, product development process has changed from being centralized, distributed, to being cooperative. The centralized approach relies on broad expertise from a few individuals; it is easy to lead but not effective for complex products. The distributed approach dispenses different product function requirements to design actors, where each individual contributes his/her expertise to the product specifications. This allows to develop the complex products but quite difficult to lead. The cooperative approach develops the product in cooperative manner among several engineers in a team work. This approach develops the products with good quality, short lead time and low cost but would confront some difficulties to solve the problems while the complexity is still high. To deal with this complexity, the system must allow different design actors to be able to communicate among each other, to discuss on the design, to negotiate and to compromise for optimizing and reducing the complexity of the design. This is what the integrated design approach does.

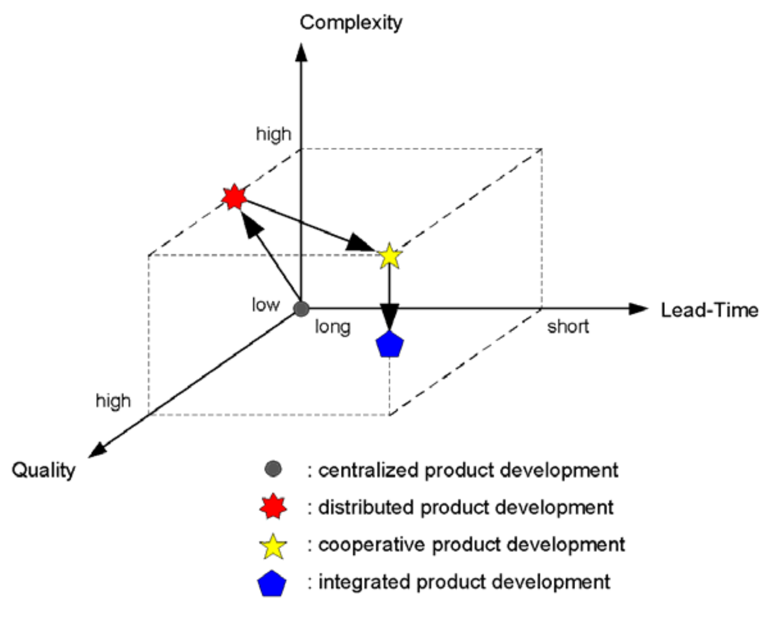


Figure 2.8 Competitiveness of product development and its evolution[8]

[8] The figure is modified from [Sohlenius 1992]

From the previous section, one can conclude that the collaborative design is constituted of: *collaborative environment* that gathers all distributed participants who intervene at any time of product life cycle into a virtual meeting room, and *decision making process* that allows participants in teams to communicate, to make decisions, and to manage conflicts during design process [Lombard et al 2005]. We propose in this study the integrated design approach that takes benefits of this collaborative design. This approach has been developed to supports the design team to integrate the knowledge from different actors in different disciplines. This proposition leads us to the problematic of this study that are:

- How can we reduce the complexity that occurs during the design process?
- How does the design team gather the information and constraints from different actors?
- How can we support the design actors from different disciplines to share their knowledge and their information?

Before answering these questions, we introduce the complexity of the design process in the next chapter. Then, we present the methods and the model for integration that are necessarily required to achieve the integrated design and also the method for solving the complexity in the second part of this thesis.

2.6 Summary

In this chapter, the development of design process has been presented. The existing and current approaches that used for developing the design process have been examined. It points out the problematic and difficulties of the design process and limitations of these existing approaches. The development of the design process aims at having better quality, lower cost, minimum lead time, and product customization.

Many researchers are attempting to develop a new approach of design process by using the emerging information technology. The rapidly changes of technology encourage the researchers to develop the design process. On the other hand, technology accelerates the expanding of globalization of both market and manufacturing. This challenges the designers to satisfy the customer's requirement of both internal and external organizations. Due to the increasing complexity of products and the shrinking product life cycle, every single design may consist of hundreds of tasks or more that are closely couples. This leads the design to be complex. To realize the nature of design, the principles of design: axiomatic design and complexity in design will be presented the next chapter.

Chapter 3

Complexity in Design

As a result of the expanding of globalization, the engineering problems today become more and more complex, especially in the area of new product development. This third chapter aims at understanding the principles of design called Axiomatic Design. It describes extensively the problematic design by introducing the theory of complexity. The four different types of complexity defined by Suh are also presented. The inherent complexity in the engineering design process is consequently examined.

3.1 Introduction

The precedent chapter has described the evolution of engineering design approaches. This chapter goes on with details of the design process by introducing the principles of design presented by Suh: the *Axiomatic Design* [Suh 2001] and by studying the *complexity in Axiomatic Design* [Suh 2003].

The nature of design is complex. [Archer 1973] wrote that “Design is that area of human experience, skill and knowledge which is concerned with man’s ability to mould his environment to suit his material and spiritual needs.” However, both of designs that rely on human experiences, skills or knowledge and that based on trial-and-error processes and empiricism seem not adequate. These experiences and knowledge must be improved by systematical approaches to solve today complexity in design. Complexity depends on the ability to synthesize. To reduce complexity, one of the goals is to replace the empirical approach with a more scientific approach. In engineering, it aims to simplify the complexity of engineering systems through the use of a rational design and systematic approach in order to reduce the cost of development and operation, increase their reliability, and enhance their performance. In manufacturing, the goal is to eliminate or to reduce the complexity while still satisfactorily remaining the function requirements of products, processes, operations, and systems under the conditions of given constraints. From the aspect of producers, the goal is to increase effectively the competition. Large companies continue to acquire smaller ones in an effort to make lower cost and to increase strategic synergies. They may employ sub-contracts or invest new plants where resources are cheap, even though the transitions create additional complexity that is contrary to the goals of previous aspects. Therefore, engineering design and manufacturing of advanced systems now require engineers, experts, and contributors who are often decentralized geographically around the world.

3.2 Axiomatic Design

[Suh 2001] has defined a definition of design as:

> "Design is an interplay between *what we want to achieve* and *how we want to achieve it*".

[Tatray 1992] also stated that design is the activity that transforms functional requirements into design parameters. As presented in the previous chapter, it might conclude that the design process begins with the perception of *needs*, continues with the formulation of a *specification*, the generation of ideas and a final *solution*, and ends with an *evaluation* of the solution. As defining the 'design process' as a 'problem-solving-process' in (2.1), we found that this conclusion is similar to the problem solving process of TRIZ (Theory of Solving Inventive Problems) that was first developed in 1946 by G. S. Altshuller and his colleagues [Domb 1997]. The schemes of TRIZ are that establishing '*specific problem*', converting the problem into a '*generalized problem*' at an abstract level, finding a '*generalized solution*' to it with reference to some known models, and then interpreting it back into a '*specific solution*' in the real situation [Nakagawa 2007]. In addition, [Suh 1990, 2001] has summarized that design process begins with the recognition of social need, formulizing of the need in a set of functional requirements (FRs), generate ideas, conceptualizing the solution, analyzing, comparing with the set of FRs to optimize the proposed solution, and checking the resulting design solution if it meets the original needs.

3.2.1 The concept of domains

To improve the design process, [Suh 2001] systematized the steps in the design process in four different kinds of design activities that founded the concept of domains. This concept made up of four domains: customer domain, functional domain, physical domain, and process domain as shown in Figure 3.1. These domains interact one another to explicit and to precise the description of the goal "what we want to achieve" that is represented by the domain on the left, whereas the domain on the right represents the design solution, "how we propose to satisfy the requirements specified in the left domain".

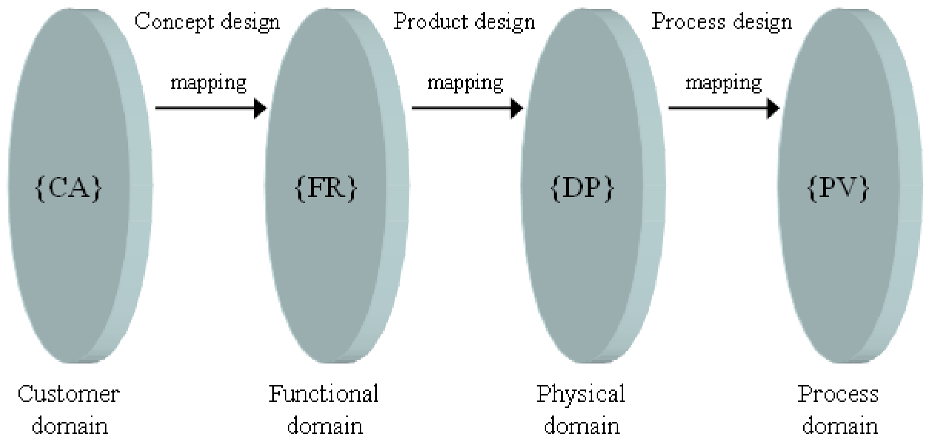


Figure 3.1 Four domains of the design world [Suh 2001]

Axiomatic Design is a scientific approach. It guides designers through the process by decomposition or mapping process. The *customer domain* represents the *customer's needs*, expectations or attributes (CAs) that for what the customer is looking for in a product. These needs and expectations are not immediately fit to be used as product specifications. Therefore, in the *functional domain*, they are formulized in terms of *functional requirements* (FRs). FRs are quantified as a set of independent requirements that completely characterizes the functional needs of the product. The mapping process between customer domain and functional domain is defined as *concept desig*n. In the *physical domain*, contents of description called *design parameters* (DPs) are created to satisfy the specified FRs. The mapping process between functional domain and physical domain is *product design*. To realize the product design, the product has to be emerged in the physical domain and be produced following the specified DPs. In the *process domain*, the manufacturing process is characterized by *process variables* (PVs). The mapping process between physical domain and process domain is *process design*. To map from one domain to another domain is called zigzagging method. Through this zigzagging method, FRs, DPs, and PVs are decomposed into hierarchies in each design domain. This process of decomposition is continued until the FR or DP is satisfied. Figure 3.2 illustrates, for example, the zigzagging method that decomposes FRs in the functional domain and DPs in the physical domain, and creates the FR and DP hierarchies. The boxes with thick lines represent FRs that are satisfied; they do not require further decomposition.

In addition, during the design process, *constraints* (Cs) are often provided into the design process. Constraints define the bounds of the acceptable design/solutions.

There are two kinds of constraints: input constraints and system constraint. Each design decision may consequently generate constraints at lower levels.

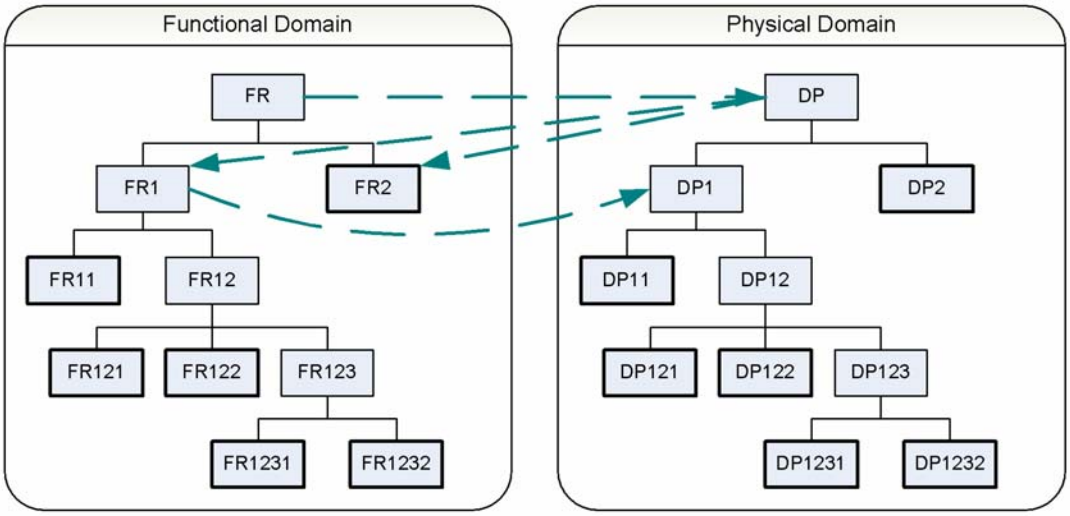


Figure 3.2 Representation of zigzagging in Axiomatic Design theory [Suh 2003]

From the concept of domains, one could notice that between product design and process design it may need a further step which includes the design and/or specification of the production system. In the process design, PVs are normally existing processes. Thus they act as constraints in choosing DPs. However, new processes may be invented for wiping out some constraints. But those new processes must take input from, and has influence on, more than a specific product.

[Sohlenius 1992] proposes an additional domain '*process function domain'* between physical domain and process domain. These five domains are separated into three worlds: *customer's world* that is *customer requirement specification*, *designer's world* that consists of functional domain and physical domain, and *manufacturing world* that consists of *process function domain* and *manufacturing domain*. To make it more clearly between the product and the process domain, *process requirements* (PRs) which represent the contents of description for the manufacturing system are required. The mapping process between physical domain and process function domain is defined as *process specification*. This concept is useful for monitoring the manufacturing environment. If there is any change in the existing manufacturing environment, the PRs need to be elaborated.

The goal of the mapping process is to define the design goals and design solutions. To sort out the alternative solutions at each level of the design hierarchy, [Suh 2001] proposes two axioms to assess the design solutions.

Axiom 1: The Independence Axiom
Axiom 2: The Information Axiom

3.2.2 The Independence Axiom

The first axiom is to *maintain the independence of the FRs*. It states that the functional requirements within a good design are independent of each other. In other words, identifying DPs so that each FR can be satisfied without affecting the other FRs. The mapping processes can be mathematically expressed in terms of characteristic vectors. At a given level of design hierarchy, the set of FRs constitutes a {FR} vector whereas the set of DPs in the physical domain constitutes a {DP} vector. The functional relationship between these two vectors is then given by an equation:

$$\{\mathbf{FR}\} = [A]\{\mathbf{DP}\} \qquad (3.1)$$

where [A] is a set of characteristics of the product design that is called *design matrix*. The design matrix [A] is of the form

$$[A] = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \qquad (3.2)$$

Each element A_{ij} represents the relation between FR_i and DP_j. In general, element A_{ij} is given by

$$A_{ij} = \frac{\partial \mathbf{FR}_i}{\partial \mathbf{DP}_j} \qquad (3.3)$$

Therefore, Equation (3.1) may be rewritten as

$$\mathbf{FR}_i = \sum_{i=1}^{n} A_{ij} \mathbf{DP}_j \qquad (3.4)$$

[Suh 1990] has separated design into three groups: *uncoupled*, *coupled*, and *decoupled*. To maintain the *Independence Axiom*, the design matrix [*A*] must be either diagonal or triangular. In the diagonal matrix, only the A_{ij} are not zero as shown in Equation (3.5). Each FR can be satisfied independently by means of one DP without effect to the others. Such a design is called an *uncoupled design*.

$$[A] = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix} \qquad (3.5)$$

In the triangular matrix, either all upper or lower triangular elements are equal zero, as shown below. In this case, the order of the DPs is the key to maintain the independence of FRs. Only the proper sequence DPs can satisfy the FRs. Such a design is called a *decoupled design*.

$$[A] = \begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (3.6)$$

The process design is characterized as well as the product design. The set of PVs constitutes a {PV} vector. The functional relationship between physical domain and process domain can be expressed as an equation:

$$\{\mathbf{DP}\} = [B]\{\mathbf{PV}\} \quad (3.7)$$

where the design matrix [B] is a set of characteristics of the process design.

Any other form of the design matrix is called a full matrix and results in a *coupled design*. Such design has many problems. To solve such problems, we have to define first the matrix [A] that solves equations between FRs and DPs. Consequently, we would solve the equations between DPs and PVs by defining the matrix [B]. Let us consider an equation in a design matrix as a task; we can introduce the three groups of design as presented in [Kara et al 1999]. They state that there are three possible task relations: dependent, independent, and interdependent task. The dependent task is presented in Figure 3.3(a), task *B* cannot be started without the input of task *A* → *decoupled design*. The independent task is presented in Figure 3.3(b), task *A* and *B* are entirely independent and could be carried out concurrently without any interaction between them → *uncoupled design*. The interdependent task is presented in Figure 3.3(c), task *A* requires input from task *B*, and task *B* requires input from task *A*. In this case, they should be carried out with iterations and negotiations → *coupled design*.

Imagine that all DPs are engaged to each FR. It is considered as a fully coupled design. If one FR is changed, all DPs must be changed. On the other hand, if DPs are deviated from the set of values, the FRs may not be satisfied. Therefore, to satisfy the set of FRs, designs must be developed and allows creating the design matrix in either diagonal or triangular form. Otherwise, the design system has to be developed to satisfy the FRs.

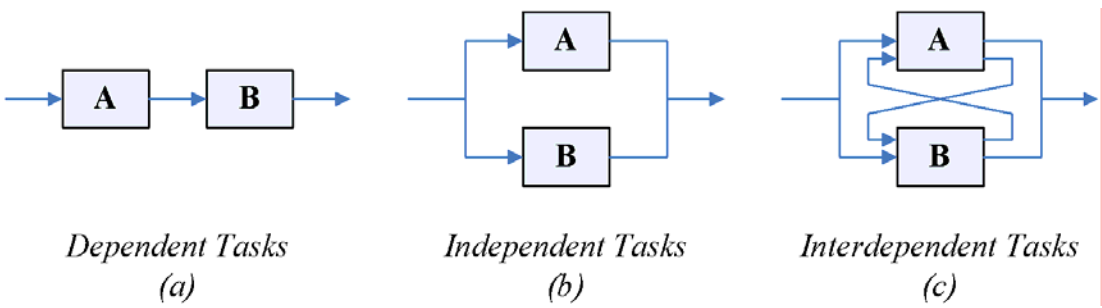


Figure 3.3 Three types of precedent relation between tasks

With regard to the whole design process, the concept design influences the product design while the product design influences the process design. In other words, FRs are engaged to DPs, whereas DPs are engaged to PVs. According to the concurrent approach, it is interesting to map the process domain to physical domain during the product design stage. With regard to *design for manufacturability*, the product design and the process design must be considered in the same time to assure that the product can be manufactured. As a result, a new design matrix [C] is a result of the design matrix [A] and [B]. The concurrent process can be expressed as

$$\{\mathbf{FR}\} = [C]\{\mathbf{PV}\} \tag{3.8}$$

while $[C] = [A][B]$ and an element of the design matrix [C] is given by

$$C_{ik} = \sum_{j} \left(A_{ij} B_{jk} \right) \tag{3.9}$$

To be able to satisfy the FRs, the design matrix [C] must be either diagonal or triangular. It is depending on the type of the design matrix [A] and [B]. Table 3.1 shows all possible results of mutual dependencies of the design matrix [C] [Lu 2006c].

Table 3.1 Types of the concurrent design matrix [C]

	Product design [A]	**Process design [B]**	**[A] × [B] = [C]**	**Result**
1	Diagonal (uncoupled)	Diagonal (uncoupled)	Diagonal (uncoupled)	Best
2	Diagonal (uncoupled)	Upper Tri (decoupled)	Upper Tri (decoupled)	Moderate
3	Diagonal (uncoupled)	Lower Tri (decoupled)	Lower Tri (decoupled)	Moderate
4	Diagonal (uncoupled)	Full (coupled)	Full (coupled)	Bad
5	Upper Tri (decoupled)	Diagonal (uncoupled)	Upper Tri (decoupled)	Moderate
6	Upper Tri (decoupled)	Upper Tri (decoupled)	Upper Tri (decoupled)	Moderate
7	Upper Tri (decoupled)	Lower Tri (decoupled)	Full (coupled)	Bad
8	Upper Tri (decoupled)	Full (coupled)	Full (coupled)	Bad
9	Lower Tri (decoupled)	Diagonal (uncoupled)	Lower Tri (decoupled)	Moderate
10	Lower Tri (decoupled)	Upper Tri (decoupled)	Full (coupled)	Bad
11	Lower Tri (decoupled)	Lower Tri (decoupled)	Lower Tri (decoupled)	Moderate
12	Lower Tri (decoupled)	Full (coupled)	Full (coupled)	Bad
13	Full (coupled)	Diagonal (uncoupled)	Full (coupled)	Bad
14	Full (coupled)	Upper Tri (decoupled)	Full (coupled)	Bad
15	Full (coupled)	Lower Tri (decoupled)	Full (coupled)	Bad
16	Full (coupled)	Full (coupled)	Full (coupled)	Bad

For the Independence Axiom, it can be concluded that each domain cannot be decomposed independently. FRs, DPs, and PVs must be decomposed into hierarchical levels by zigzagging between the domains until the design is completed.

3.2.3 The Information Axiom

The second axiom is to *minimize the information content of the design*. Information is also related to the notion of complexity. Therefore, at each hierarchical level, designers must choose a minimum number of FRs. The design with less information content and still satisfying the FRs is better. In the case of developing an existing product, one effective tool that is widely used to formulate customer needs for achieving FRs is that *House of Quality* (HOQ) also known as *Quality Function Deployment* (QFD).

Rationally, designer determines a design in terms of the probability of achieving the design goals. The information axiom states that the design with the highest probability of success is the best design. Information content is a measure of the *probability of success* (P) of achieving the specified FRs (for *product design*) or DPs (for *process design*). The probability of success is the function of *design range* (dr) and *system range* (sr). The *design range* is the specified FRs, whereas the *system range* is the capability of the proposed solution. The overlap area between dr and sr is called *common range* (cr), $cr = dr \cap sr$ as shown in Figure 3.4.

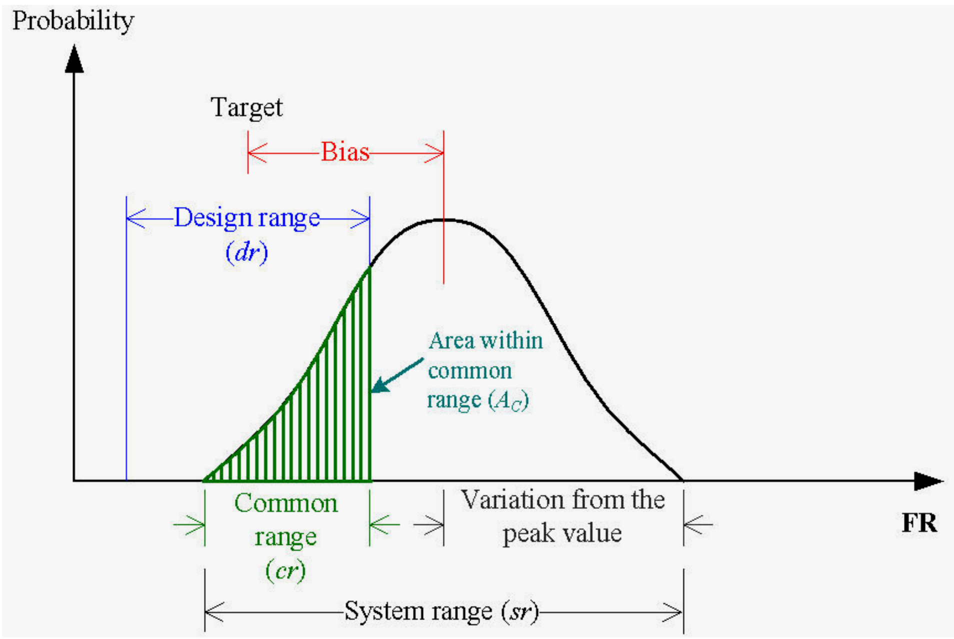


Figure 3.4 Design, system, and common range

In Axiomatic Design, information content is defined as a logarithm function of P of achieving the specified FRs that can be written as

$$I = \log_2\left(\frac{sr}{cr}\right) \qquad (3.10)$$

If the whole area of *dr* is a part of *sr* (*P* is equal 1.0), the information content is zero, on the contrary, if the whole area of *dr* is outside of *sr* (*P* is equal zero), the information required is infinite. That is, the less probability, the more information required to satisfy specified FRs. Then, the total information content of the system is obtained by summing up individual information content which corresponds to a set of FRs. Normally, outputs of the design process are *information* in the form of drawings, specifications, tolerances, and other relevant knowledge. Thus the design solutions should be as simple as possible; the total information should be smaller as possible, so the design output can be transmitted with minimal effort.

In conclusion, one might notice that Axiomatic Design's philosophy is similar to TRIZ. Axiomatic Design uses a "solution-neutral environment". It means that FRs must be defined without ever thinking about something that has been already designed or what the design solution should be. TRIZ uses the methodical thinking process as one said "thinking out of the box". Instead of spending time for problem definition and fumbling a solution for those problems of a system or product, TRIZ looks at the most ideal state of that system or product. Then find the contradictions and resolved for reaching that ideal end state by using existing tools and resources. In addition, Axiomatic Design has design rules (corollaries and theorems) to develop and determinate a design, whereas TRIZ has inventive principles and problem solving tools as guidelines for solving. Many design rules of Axiomatic Design and problem solving tools in TRIZ are related and share the same ideas in essence. [Mann 1999a, Mann 1999b, Yang and Zhang 2000a, and Yang and Zhang 2000b] have reviewed and analyzed these two methodologies. They conclude that the basic foundations of Axiomatic Design and TRIZ can enhance each other to solve the problem, aid to come up with design concepts, make the design process clearer.

3.3 Complexity in Axiomatic Design

The term "complexity" is commonly found in use throughout all fields of science including physics, biology, sociology, etc. It is not surprising that there is no homogeneous definition of complexity. [Gershenson and Heylighen 2005] describes the complexity as: distinct components that are joined and mutually entangled, a change in one component will propagate through a tissue of interactions to other components which in turn will affect even further components, including the one that initially started the process. However, to answer precisely the question – what is the complexity? We need first to clarify the object of the question – "complexity of what?" This section presents the complexity in engineering from the perspective of Axiomatic Design which is presented in the precedent section. Although the term 'complex' is often considered as a synonym of 'complicated' just as they are in

English dictionary[9]. In engineering design, a complicated object may be decomposed into elements which can be resolved and recomposed as a simple object, while a complex object can be decomposed into elements as well but it may not be resolved the complexity. However, increased complexity may not mean more complicated or difficult to use a product or a system. It requires, of course, more engineering knowledge and skills to develop complex products and processes but it aims at easier use such those products and processes.

[Suh 2005] shows an example of typical engineering problem that depicts the notion of complexity. Consider the task of cutting a rod to 1 meter. The complexity of this operation depends on the accuracy of the rod that has to be cut. If the FR is to cut the rod within m. or cm. it can be done easily; therefore it is not complex. On the other hand, if it requires being cut within mm. it is difficult to achieve; thus it is a complex task. [Sohlenius 2004] said that "Complexity must be combined with simplicity through good engineering. Especially skill in dealing with uncertainty in the design of complex products is important". He further commented that "Simplicity means high probability to succeed, which is the same as high probability to meet all the defined functional requirements within tolerances, that is to say low uncertainty". This comment agrees with [Suh 2003], who has defined complexity as

> "Complexity is a measure of uncertainty in understanding what it is we want to know or in achieving a functional requirement (FR)."

According to the Axiomatic Design, complexity is related to information: the more complex a product or system is, the more information required. [Suh 1990] defines information as a logarithmic function of the probability of achieving the specified function requirements. He also summarizes that complexity arises when we cannot give a complete description to a product or a system. As shown in Figure 3.2, each FR and DP is decomposed into hierarchic levels as branches. Refer to the three possible task relations of [Kara et al 1999]; each element of each branch (a highest level of FR or DP) has a relation to each other. If designers do not understand the behavior of each individual (lacking of understanding or knowledge), the complexity arises when the branches have been merged at higher levels.

[Suh 2001] has classified complexity into two kinds: *time-dependent complexity* and *time-independent complexity*. Time-independent complexity is further divided into time-independent *real complexity* and time-independent *imaginary complexity*, depending on its root cause and does not require time dimension. On the other hand, time-dependent complexity involves time as one of its determinants. It is also divided

---

9 *Complex*: Consisting of parts or elements not simply coordinated, but some of them involved in various degrees of subordination; complicated, involved, intricate; not easily analyzed or disentangled.
Complicated: Consisting of an intimate combination of parts or elements not easy to unravel or separate; involved, intricate, confused. From Oxford English Dictionary 2nd edition, 1989

into two different types: time-dependent *combinatorial complexity* and time-dependent *periodic complexity*.

3.4 Time-independent complexity

Time-independent complexity is the complexity where a system range and uncertainty of achieving the functional requirements does not change over time. Since the uncertainty is related to the set of FRs, it can be concluded that functional requirements are also time-independent. This kind of complexity is embedded in the design itself. To reduce the time-independent complexity, *we must first have a systematical design process*. To have such the design process, *the time-independent imaginary complexity must be eliminated*. As a result, we can reduce the time-independent real complexity that remains.

3.4.1 Time-independent real complexity

Time-independent real complexity is related to the uncertainty of a system. [Suh 2003] defined real complexity as "*a measure of uncertainty when the probability of achieving the FR is less than 1 because the common range is not identical to the system range*". [Lee 2003] redefined this definition and stated that real complexity is *'the complexity caused by system range's being outside of the design range'*, (see Figure 3.3). This definition implies that the uncertainty exists even the independence axiom is satisfied, as long as the common range is not the same as the system range. To determine the real complexity, we need to establish first the design range of the FRs. Then state the constraints at each level and establish the system range following the DPs. The real complexity (C_R) can be computed by determining the overlap between the design range and the system range as illustrated in Figure 3.4.

Note that as long as the design does not change, the system range is not going to change. Therefore, the real complexity will not be reduced. For uncoupled design, the real complexity may be reduced by changing/adjusting the corresponding DPs of each FR until the system range overlap to the design range at most as possible or make the design range larger. Decoupled design is as same as uncoupled design but changing the DPs must be in the sequence given by the design matrix, since the change of each DP affects to the other DPs. The more difficult case of reducing the real complexity is when the design is so fully coupled that the bias cannot be removed since FRs are dependent on each other. In this case of a fully coupled design, even stiffness of an FR cannot be reduced lest it adversely affect the stiffness of other FRs. In this case, the best way is to develop a new system to replace the coupled design with an uncoupled or decoupled design.

The uncertainty of the system is represented by the deviation of FRs that may arise from the variation of design parameters, design matrix, or noise factors. To deal with the real complexity, [Lee 2003] proposed three approaches which based on technical and economic consideration, and must be combined for optimal result.

- Eliminate the source of variation: is to identify the root cause of the variation, and then reduce or eliminate the source by using methods such statistical process control (SPC) and mistake proofing (Poka-Yoke).
- Desensitize the system: is to minimize the output variation by making the system insensitive. This approach uses methods such robust design, also known as Taguchi method.
- Measure and compensate: is to measure the deviation and then find some parameters '*compensators*' to cancel the effect from input variations and noise factors.

We can find some case studies of how to reduce/ eliminate the time-independent real complexity have been presented in [Suh 2003] e.g. reduction of time-independent real complexity of a knob, an injection mold, an internal combustion engine, etc.

3.4.2 Time-independent imaginary complexity

Although a design satisfies both of the independence axiom and information axiom, uncertainty may still exist; this uncertainty is called imaginary uncertainty. Time-independent imaginary complexity is caused by lacking of knowledge and proper understanding of designers in a specific design and system. When the design is uncoupled (a diagonal matrix) as illustrated in Equation (3.5). There is no imaginary complexity because the design can satisfy the FRs in any order. Equation (3.11) shows by example a decouple design structured with *m* FRs and *n* DPs as a triangular matrix, where m = n. An *X* in the design matrix indicates that there exists a functional relationship between a DP and a FR. This design satisfies the independence axiom. There is no real uncertainty associated to it as long as the DPs are changed in the indicated order and each system range is inside the corresponding design range.

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ \cdots \\ FR_m \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & \cdots & 0 \\ X & X & 0 & \cdots & 0 \\ X & X & X & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ X & X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ \cdots \\ DP_m \end{Bmatrix} \quad (3.11)$$

However, it may be a source of imaginary complexity if the designer does not recognize that is a decoupled design. As a result of the trial-and-error approach, there are $n!$ different sequences of DPs, of which only one is correct. Thus the probability of finding the right sequence of n DPs to satisfy the m FRs is given by

$$P = \frac{1}{n!} \tag{3.12}$$

If n is 5, the probability to find the right sequence is 0.008, which is very small. That is to say, the more number of DPs is the less probability of finding the correct sequence. Since the probability is very small, the uncertainty is large; the design is then considered complex.

[Suh 2003] shows a simple example of the imaginary complexity that is: assuming that every professor in the Department of Mechanical Engineering at MIT agrees on FRs and that the FRs can be satisfied independently. However, each one of the 60 professors in the Department has different views on the best DP that can satisfy the FR. Because of this diverse opinion of the faculty, the decision making can be complex if each one of the professors can affect the outcome, since the FRs may not be satisfied within the design range. What is the best decision making process that will enable the Department achieve the FR?

The simple and the best solution is that gathering all the opinions of the faculty and understand their implications and then the person in charge (normally the Department Head) make the final decision to be sure that the uncertainty and thus, the complexity is minimized.

In fact, this imaginary complexity is generated by the designers themselves, as a result of not knowing/understanding the exact relationship between the FRs and DPs of the system. This leads to wrong decisions of choosing parameters and increases the uncertainty, and consequently the design is defined complex although it may be not. To deal with this kind of complexity, it will be discussed in Chapter 5.

This section has discussed the time-independent complexity involved in making design decisions. The real complexity is associated to the uncertainty that is inherent in the system. This kind of complexity exists when the system range is outside of the design range. The imaginary complexity is associated to the uncertainty that is a resulted by lacking of knowledge and proper understanding or wrong choice of design parameters.

3.5 Time-dependent complexity

Time-dependent complexity is contrary to time-independent complexity as its name defined. This kind of complexity occurs because future events affect the system in unpredictable ways. For time-dependent complexity, the uncertainty changes as a function of time. The uncertainty changes can come from either *time-varying system* *range* or *unpredictability of functional requirements in the future*. This often results in the time-varying system range, that is, the system range moves away from the design range as time goes on (see Figure 3.5). This varied range causes the system unreliable. The important mean is to reduce the time-dependent complexity to increase the reliability of the system. Time-dependent complexity is divided into two different kinds: combinatorial and periodic complexity.

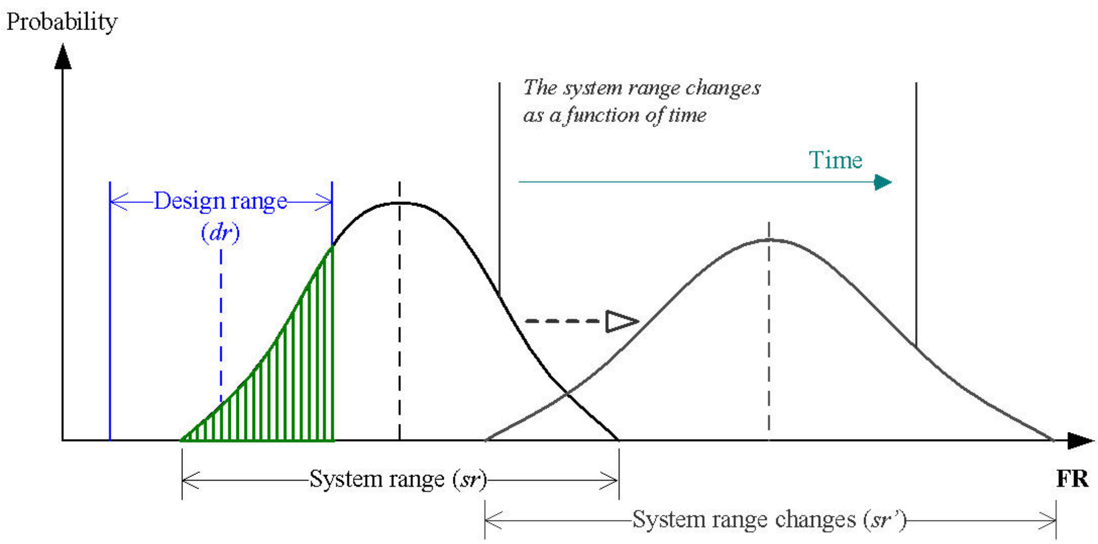


Figure 3.5 System range changing as a function of time

3.5.1 Time-dependent combinatorial complexity

The combinatorial complexity arises when the system range changes as a function of time and moves away from the design range in an unpredictable way. The uncertainty variations arise not only because of the affects of future events but depending on the decisions made in the past as well. A scheduling problem can exemplify the combinatorial complexity. For example, to schedule a job shop, the scheduler must deal with varied machines and varied parts which are brought from different (internal) customers or departments. Selecting of which parts are produced using which machines in the earlier is affected to the future scheduling. Any delay brought of any customer also affects to the schedule of others parts or machines. Another example is the airline schedule in bad weather as explained in [Suh 2005].

A system with combinatorial complexity is most likely to fail because of a long time or infinite time running period. To reduce such complexity, the system must perform in a predictable way by transforming the combinatorial complexity into time-dependent periodic complexity. This could be done by introducing functional periodicity to make the system more stable and reliable. Suh proposes some of the functional periodicity as following types:

- Temporal periodicity
- Geometric periodicity
- Biological periodicity
- Manufacturing process periodicity
- Chemical periodicity
- Thermal periodicity
- Information process periodicity
- Electrical periodicity
- Circadian periodicity

Such functional periodicity has been described in [Suh 2003].

3.5.2 Time-dependent periodic complexity

The periodic complexity is similar to the combinatorial complexity but having a finite time period. Take an example of a schedule of bus or train in France or elsewhere. The schedule begins at the early morning of each day and ends lately at night. The schedule starts over every day. If there is any unpredictable event or accident, the schedule might not be on time as it should be, in other words, the system range would move away from the design range. However, the schedule can re-run in the next day and resume the regular schedule (temporal periodicity).

A coupled system can be changed to a decoupled system since a set of FRs repeat periodically. With the recurrence of a set of FRs, the system can reinitialize itself over each period. The system is then stable and reliable.

This section has discussed the characteristics of time-dependent complexity. To reduce the time-dependent complexity, the system must perform in a predictable way. When the system range is stable and reliable, we can adjust the variations, parameters in order to make the system range overlap to the design range as the real complexity does. Therefore, both of combinatorial complexity and periodic complexity are considered real complexity.

3.6 Summary

This chapter has examined the Axiomatic Design and the theory of complexity. Axiomatic Design approach employs the concept of domains that systematizes the design process in four different domains. Axiomatic Design consists of two axioms that rule the design process: the *independence axiom* and the *information axiom*. The independence axiom decomposes the design process in hierarchies by zigzagging between those four domains until the design is complete. To maintain the independence of FRs, the design must be either uncoupled or decoupled. The information axiom minimizes the information content of the design in order to achieve the design goals which are represented by a set of FRs. The information content relates to complexity. Since the probability of satisfying the FRs is small, the design is then considered complex.

The theory of complexity characterizes the complexity into two groups: time-dependent complexity, and time-independent complexity. To reduce the complexity of design, the objective is to lay the system range in the design range. It can be done by either move the system range closer to overlap the design range as much as possible or make the design range larger to lie over the system range. This study is interested in the time-independent imaginary complexity that ought to be reduced prior to the other complexities. This kind of complexity often arises when we have to satisfy many FRs in the design process. A method for solving the imaginary complexity is continually discussed in Chapter 5.

Part II

Toward an Integrated Design Approach

Chapter 4

Models and Methods for Integration

The fourth chapter introduces the concept of integrated design that has been developed beyond the existing approaches presented in the second chapter. It is essential that the design actor have to contribute their knowledge and constraints as soon as possible during the design process. Gathering together the information from different disciplines is complicated. This chapter presents methods and models for integration, multidisciplinary concept that allow the designers to gather and to integrate that knowledge to perform design activities.

4.1 Introduction

The fundamental issues that must concern with the collaborative design approach are *collaborative environment*, and *team decision making*, as presented in Chapter 2. To develop the integrated design process, only these two issues are not adequate. We must address *knowledge management* issue that enhances the designers to achieve the knowledge integration. This chapter aims to develop the integrated design that employs previous studies developed by the "integrated design" team of laboratory G-SCOP (3S). The history began in 1991 by [Belloy 1994] who inaugurated a new design process approach. His study articulates on the formalization of knowledge, rules of production, and notion of entity, that are the know-how of design actors. The objective of the study is to be able to integrate the manufacturing process into the design process as soon as possible.

[Chapa Kasusky 1997] aimed to propose a methodology for integrating the different actors who get involve during product's life cycle, and to establish the tools that permit the design actors to cooperate in the context of integration. To achieve the objectives, Chapa Kasusky implemented the notion of holonique design by established the concept of *product model* and formulated the rules of this model using the association of components, links, and relations. This study also proposed the concept of *internal actor*, *multi-actor*, and *common views*.

[Mer 1998] observed common characteristics of the design process and developed tools to support the design activities. This study postulates that one designer is an expert in his/her competence but has different comprehension and roles in the design process. The '*concept of world*' is proposed to collect heterogeneous information from different competences by exploiting the concept of product model. The '*concept of translation*' is also proposed to associate with the inherent information between different *worlds*.

[Roucoules 1999] continued to develop the notion of entity and the notion of product model. This study has accomplished to associate *knowledge model* to *data model*. This task permits the designers to extract their knowledge and constraint, into the design process, in the form of features – characteristics, behaviors. – of the product. An integrated design modeller and design tools are realized and also the concept of *multi-view* and *multi-representation* are invented in this study. The design actors are permitted to participate to the design project by their own platform (Windows, Linux, Silicon, etc.) and could exploit their specific tools to evaluate the design.

These previous studies have created a design process approach with methods and tools based on the context of integrated design, and have developed an integrated design tool named "CoDeMo", Cooperative Design Modeller [Roucoules and Tichkiewitch 2000]. CoDeMo has been employed to validate the created methods and tools. This study continues to develop the notion of product model, formalization of knowledge of different trades. CoDeMo is then implemented to manage the interaction among design actors from different competences.

4.2 Models for integration

Designers normally record information results with reasoning and calculations in a private notebook which is not easily to be shared. Although design information is recorded in the form of text and graphics, which can be captured electronically, much of the design intent in the form of dialog and face-to-face interaction is lost [Sky and Buchal 1999]. In addition, [Heylighen 2002] states that "*the explosive development of the internet and related information and communication technologies has brought into focus the problems of information overload, and the growing speed and complexity of developments in society. People find it ever more difficult to cope with all the new information they receive, constant changes in the organizations and technologies they use, and increasingly complex and unpredictable side-effects of their actions*". Therefore, the question is how to manage the explosive information that includes both of relevant and irrelevant information? This issue needs to be addressed and included into the design process. Therefore, this section presents how the product is modeled, how to structure the product data for integration, and how to capitalize the knowledge of the design actors.

4.2.1 Product model

Formerly, product model was supposed to describe mainly geometrical data. For example, a drawing file created by a CAD system mostly contains geometrical data that implies only dimension and specifications. In this case, the notion of product model following the study of [Chapa Kasusky 1997] does not concern only the geometrical data but also means to information which comes out during the design process to complete the product. The product model in the context of integration is a model of informatics that is constituted by associating *knowledge model* into *data model*. Product model is comprised of knowledge model and data model. *Data model* is considered as a structure of product model. It consists of *component*, *link*, and *relation* while *knowledge model* consists of *factual knowledge* and *temporal knowledge*, as shown in Figure 4.1.

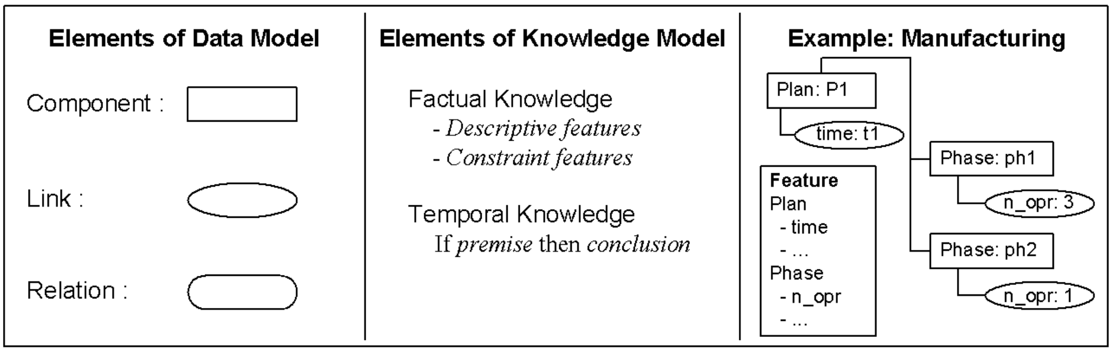


Figure 4.1 Product model associated between data model and knowledge model

<u>Knowledge model</u>

Knowledge model allows the design actors to project their knowledge or to define relative information to the product during the design process with their own comprehension, vocabulary, and manner. However, regarding to the context of integration, some parts of this knowledge pertaining to more than one actor must be shared and used with other trades. Therefore, to facilitate the design actors to comprehend such knowledge of each other, a method for translation must be provided. This method will be presented in the next section. The knowledge model can be characterized into two categories: factual knowledge and temporal knowledge.

Factual knowledge is represented by "*feature*". Initially, features were mainly used to describe geometrical data using in CAD systems. In this study, we enlarge meaning of features to express as well information in other domains. A feature can be defined by a name, characteristics, and behaviors given by its users, depends on the context in which it has been created. Features may have a same name in *taxonomy*[10] but they are in different *ontology*[11]. A same name of feature can be used in different context. For example, a *keyboard* using with a computer has a different meaning to a *keyboard* which is a musical instrument. Therefore, one feature specified by an actor is independent from others. Nevertheless, values of characteristics of feature can be affected by *temporal knowledge*. A feature is an object manipulated by design actors, which describes the product. In the integrated design methodology, we can further divide *features* into two categories as following [Roucoules and Tichkiewitch 2000]:

- *Descriptive features* describe the product with specific vocabulary according to a specific trade's point of view. For example, a *Cylinder* feature describes a cylindrical form of the product with its characteristics i.e. radius, length, and area. A behavior of this feature could point that the area value is linked to the value of radius and length. Note that a descriptive feature can be tangible or intangible. For example, geometric form features such as *Cylinder*, *Rectangle*, *Circle*, etc., *are* tangible features using in geometric view while manufacturing features such as *Cutting*, *Drilling*, *Milling*, etc., are intangible features using in manufacturing view.
- *Constraint features* are used to define constraint on descriptive feature characteristics. For example, Equality is a constraint feature that is defined with two characteristics: variable1 and variable2. Its behavior imposes these two characteristics must be equal.

Temporal knowledge is represented by "*production rules*". A production rule begins with a premise and finishes with a conclusion such as: If *premise* then *conclusion* [Tichkiewitch 2002]. The premise is a logical proposition taking into account the state of one or more characteristics of features. The conclusion may create an instance of a feature, define some values of characteristics, or start a specific procedure. This temporal knowledge enhances the design actors to share and to exchange their information in the team.

The knowledge model can be enriched by the interaction between the design actors that will be presented in the next chapter. In Chapter 6, we describe more details about the knowledge model and present how to constitute the knowledge model in trade views.

---

[10] *Taxonomy* is the science of classification according to a pre-determined system, with the resulting catalog used to provide a conceptual framework for discussion, analysis, or information retrieval. [WhatIs.com 2005]

[11] *Ontology* is an explicit specification of a conceptualization, used to help programs and humans share knowledge. [Gruber 1993]

Data model

The design actors cannot define their knowledge model such characteristics, behaviors, or values to the product without data model. The data model composes the structure of product model and is associated with knowledge model. It is considered as a skeleton which stores coherent descriptions of product. We define three types of object: component, link, and relation; that formulize the data model as shown in Figure 4.2. This section describes moderately about the concept of data model, however the more details can be found in [Chapa Kasusky 1997] and [Tichkiewitch 2002].

A component represents the description of a product. It may describe physically a part, a set of parts, or a portion of part. It also can be a material set, a temporary element before manufactured, depending on the actor in a specific trade view. A component is an instance of a feature and its characteristics.

A link is associated to a characteristic of a component or an association of characteristics which it addresses. As name defined, it is used as a connecting node between components.

A relation represents a connection between two links or more, which are of the same component (this is called 'behavior') or different components. A relation adds a constraint feature between two links or more.

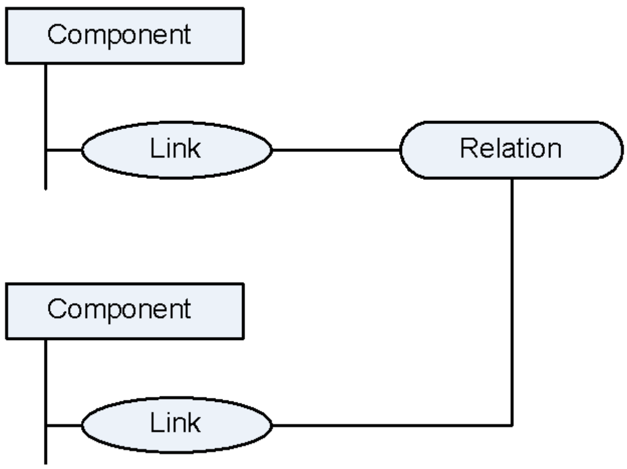


Figure 4.2 Graphic symbols and formalization of data model

4.3 Methods for integration

In any design process, ideally, all relevant information, knowledge, and constraints from all contributors should be brought together before making a decision. However, under the condition of globalization, design teams, experts, and/or contributors have been decentralized geographically in different locations. This leads them to confront some difficulties in communication, sharing/exchanging information and knowledge. Furthermore, obtaining pertinent, consistent, and up-to-date information across a large company is complex and time-consuming. [Poolton et al 2000] states that with such problem, around one-third of new products will fail to meet their targets margin due to the ineffective exploitation of expert knowledge. This knowledge is often fragmented and is difficult to achieve the capitalization on critical success factors. This section presents then the concepts and methods that facilitate the design actors to organize the design activities, and also the methods for integrating knowledge and know-how of different trades into the design process.

4.3.1 Integrated design concept

Integrated design focuses on knowledge integration during the design process and supports designers to achieve the collaboration of design activities. [Roucoules et al 2003] presented two manners to perform knowledge integration, i.e. integration and distribution.

Integration – One objective of the integrated design approach is to reduce design iterations during the design process. These iterations resulted by the delayed or non-integration of knowledge between design department and other departments such as manufacturing. Integration aims to gather relevant information of the whole product life cycle as much as possible. Instead of waiting until a conflict happened, every design actors must participate to the design team and describes their points of view or any potential problem to the team. A design actor has to say as soon as s/he can say but only s/he can justify it. As a result, each one can formulize the problems or any coupled phenomenon in the design.

Distribution – In the design team, each design actor normally has competence only in his/her domain. Thus, they might not provide any information outside of his/her experiences. However, between each design actor, there is some coherence information that they interest in. Therefore, one's knowledge must be distributed to another one(s) who interest in. This enhances them to understand each other and to be able to formulize the potential problems and coupled phenomenon during the design process.

Therefore, integration and distribution are inseparable. The crucial factor of integrated design is how to make the team to be able to communicate to each other during the design process. [Gaucheron 2000] characterizes the notion of integration in three significations: interconnection, coherence, and interaction.

- *Interconnection* gives heterogeneous computer system a connection to be able to communicate to each other with geographical distant and exchange rapidly information.

- *Coherence* gives design actors a distant access to a shared database. It permits the design actors to access, to share, and/or to modify shared information but only information which they concern to.
- *Interaction* – the idea is that each design actor, who is in charge of any part of product life cycle, has to provide prior his/her constraints and knowledge to the project. This contribution gives designers gathering problems as soon as possible in order to solve problems at the early of design process.

The integrated design concept aims at integrating all the knowledge in product design as much as possible. Indeed, this integration reduces the number of design iterations and, consequently, the design time. The problem we have to solve is how to permit the design partners to communicate each other. Therefore, the industrial aim is to have remote formal and informal communications instead of gathering them all to be presented at the same place, which costs expensive and might not be possible. From the general concepts of CAID (Computer Aided Integrated Design) system as presented in [Tichkiewitch 1996], CoDeMo has been developed to supply a formal level of communication.

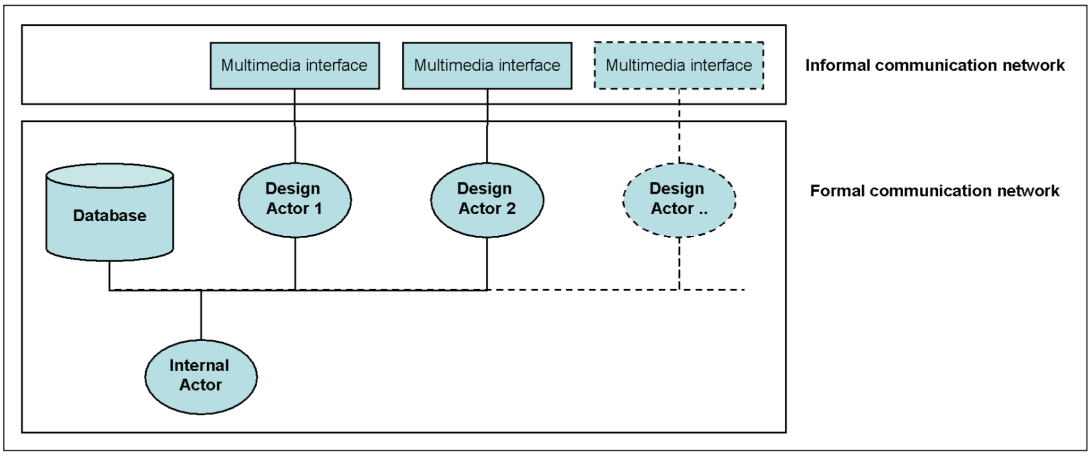


Figure 4.3 Concept of knowledge integration

Figure 4.3 shows the concept of the integration via a formal and an informal exchange of data. If one design actor works alone and never share his/her results or problems, s/he will never be integrated in the design project. The design actors would rather share and post their problems to the team. In this way, a formal exchange of data is required. In CoDeMo, a shared database is the central communication point.

The first step of the integration is that the actors create new data by retrieving needed information from the initial data and then evaluate step by step the design. The results from the evaluation are new data of the product. For example, to plan a manufacturing process of a part, the geometric data, assembly solutions and tolerances are required. Then, the manufacturing process can be chosen (drilling, milling, stamping, etc.) and manufacturing parameters (feed and cut speed, etc) can be evaluated. CoDeMo permits each design actor to add, edit, or modify the information via the graphical user interface (GUI) to the database. They can also use a neutral file which is compatible with many applications and is automatically generated from the database. However, the input and output formats have to be known and these are not always obvious. Such a STEP format, for example, a standard for the exchange of product model data, is now reliable in CAD systems for geometric data.

The second step of the integration is to send the new data from the evaluation to the shared database. This step allows the design actors to integrate their own information and knowledge into the project. That information will either increasingly define or constrain the product according to relations between the new and existing data. For instance, the mechanical expert can define the minimum thickness of a part that consequently constrains the maximum hole at the edge of the part.

CoDeMo enables the informal communication channels that permit the design actors to communicate to one another. In fact, the concept of CAID consists of more details. The structure and architecture of CoDeMo will be presented in Chapter 7.

4.3.2 Integrated design method

In the context of integrated design methodology, the design process can be divided into two phases of integration. During these two phases, the designers handle the product model to deal with the knowledge integration and the mapping product functions to product structure.

First design phase – Following the study of [Belloy 1994], we ask the designer, who is in charge of the global form and the esthetics of the product, to transform the product specifications into a conceptual product model. In order to recognize the functional surfaces of the product, we ask the technologist to manipulate his/her knowledge into a product according to product's main functions. This is to provide initial information and to facilitate others actors to recognize the functional surfaces of the product. In this study, we initiate a design project by transforming an exported file from a CAD system. This is presented in the next chapter.

Second design phase – From the information provided in the initial design phase, other designers are able to provide their information/description by adding new data, modifying, and correcting existing information. They can also bring their constraints into the design and allow the team to choose an available solution. Otherwise, if there is not an available solution, the team has to find/create a solution to resolve that problem. This collaborative work can be done by knowledge integration.

These two phases are not actually separated as the systematical design approach (see 2.2) but rather is a progressive integration of designers during the design process as shown in Figure 4.4. The overlapping area is a database that stores gathered information, knowledge, and constraints of all design actors. This overlapping phase must be treated simultaneously as soon as possible between the first design phase and the second design phase. It is certain that the design process starts with the first design phase and will be finished in the second phase.

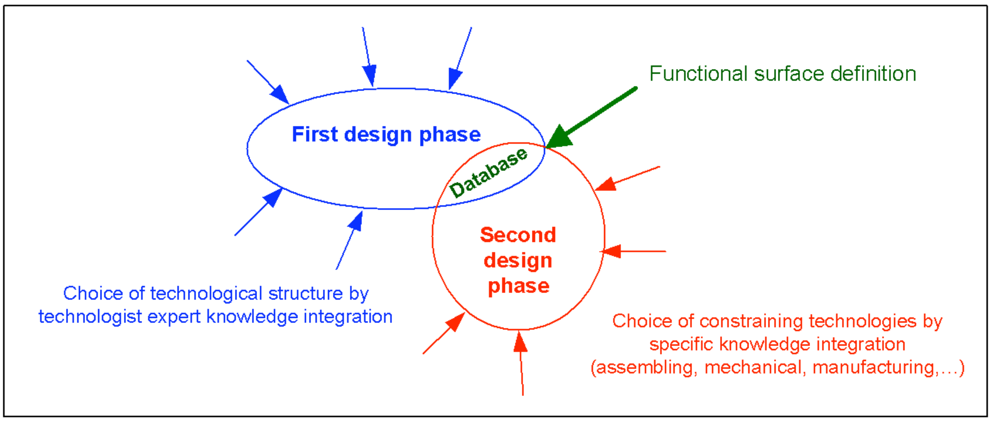


Figure 4.4 First design phase and second design phase

With the knowledge integration, the design approach is changed. Integrated design attempts to avoid lacking of information by bringing firstly the relevant information and constraints, all the product life cycle, as much as possible to formulize the geometrical model of the product. This approach uses no more the product geometry at the beginning of project as traditional CAD model. The result of integrated design method provides the geometry of product (see Figure 4.5).

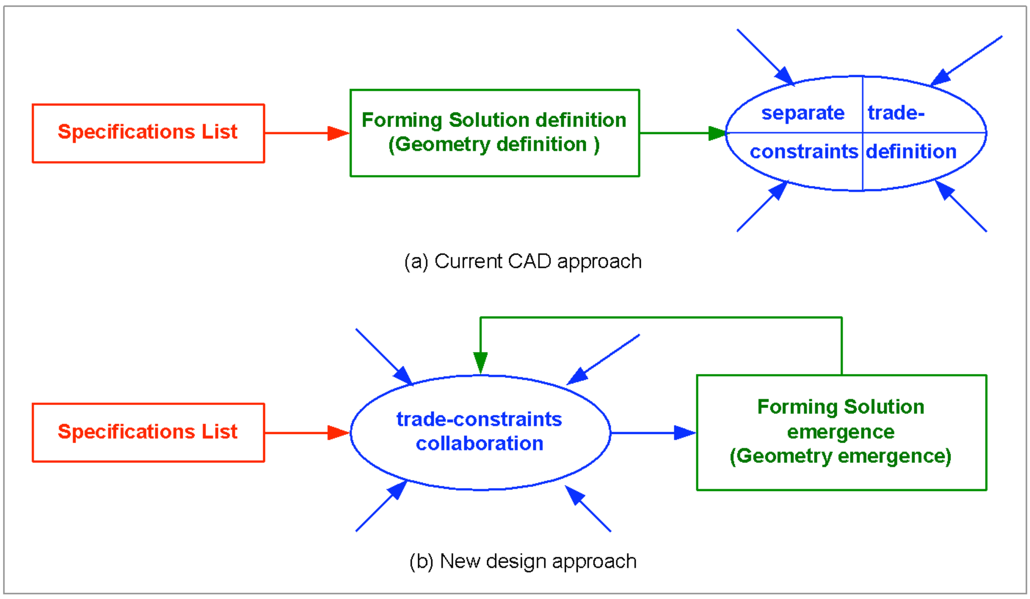


Figure 4.5 New approach of product design method

4.3.3 Collaborative environment

Network based approach has been employed in communication since a long time. Numerous researches benefit this approach to introduce a collaborative environment. Collaborative environment is used to solve time delay problems which resulted from asynchronous communication. The time delay problems consequently result delayed product development and lead to lack of ownership of design decisions. It is used to support the design team to perform the design activities by providing a balanced solution for multidisciplinary concept, sharing data and resources between design actors and also for the knowledge integration.

Collaborative environment facilitates the design team to be able to communicate to each other, to exchange information and knowledge, or to present their constraints into a virtual meeting room. However, [Huang 2004] states that the design process is often a conceptualization, which is not easy to share and is seldom documented formally. Otherwise, some intent information may be lost. As a result, a complex design is often carried out through collaborative works. Furthermore, different disciplines concern different objective but must be integrated to achieve the common goal. This is a reason why the integration takes the important role in this part of the design process.

4.3.4 "Worlds of design" concept

[Mer 1998] presents in his study that each design actor does not have the same symbolic systems, representations of product, or evaluation tools, etc. He elaborates the concept "*worlds of design*" to realize this phenomenon. This concept signifies that an actor can have different manner of apprehending on the same conceived object as a result of standards and tools which s/he is using for evaluation. Mer defines the concept of world as "*A world of the design is a group of heterogeneous entities (which can be tools, objects, persons) that develop the same logic of action, raise the same scale of size, and share collective knowledge*". These three notions constitute the concept of worlds. The notion of '*logic of action*' associates the objective of actions, the constraints, and the values to the actions, which affect to the product. It signifies that there is continuity between every actions of an actor. The notion '*scale of size*' is associated with the logic of action. It permits to legitimate the actions (why we do this), the tools (why we choose this tool), and the objects (why we use this operation). It is not a 'principle of justice' but a 'principle of reasoning' (appropriateness of the action) that guides the actions. The last notion '*collective knowledge*' means knowledge, conventions, or implicit/explicit rules, which are shared to every actor in a world. Therefore, it is essential to acquire the shared knowledge of the actors of this world. (It can be summarized that '*world*' signifies to the design process.)

4.3.5 Intermediary object

[Suh 2001] stated that "complexity arises when we are unable to deal with or understand the behavior of the aggregation of individual elements". During the design process, numerous omnipresent communications and outcomes are generated by the design actors. This may raise complexity in the design process and mislead design actors to wrong direction. Therefore, giving "*common object*" is a role key that supports the design actors to perform design tasks with the *coincident notion*. In other words, "*common object*" is called "*intermediary object*" that makes them understanding each other during the design process.

[Mer et al 1995] defined the intermediary object as an important role of communication in the design process, not only to support information but also as an instrument of coordination between the design actors. His study proposing the hybrid nature of intermediary object is presented with two aspects that are inseparable: first aspect, it is as a representation model of future product. This representation is contextual and relative to the knowledge that constructs the product. It also represents the process that it is a result of. The second aspect is an instrument of coordination or cooperation for the design actors. The people who have the same interests can use the same objects. This is to decrease and to group the divergence during the design process. They are such vectors of communication for different competences and design actors.

The intermediary object can be characterized as *messenger* or *mediator* object. The messenger object is a transparent object which transmits an intention or an idea of its producer (a user/design actor). It does not modify any intention even the intention or the idea is deformed. On the other hand, the mediator object can modify the initial intention in the comprehensible form before sending it to the receiver. It interposes between the idea of sender and the usage of receiver. It is an internal actor (4.2.6). In the meantime, the intermediary object can be characterized as *opened* or *closed* object. The opened object gives the users (design actors) latitude which can be more or less divergent. This object generally concerns the interpretation tasks. On the other hand, the closed object decreases the divergence and gives the users only relative context. This object transmits principally a prescription such as a manufacturing plan which is mostly concerned by the manufacturer. However, in order to integrate points of view of different trades in the product life cycle, the object must be opened as much as possible.

4.3.6 Multidisciplinary concept

The globalization has challenged the design team to develop the design process to satisfy the customer's requirements with given criteria of quality, cost, and time. As presented in the second chapter, one of the most critical issues of design process is that the designers have to work in collaboration as teamwork. It means that they must be able to dialog, to discuss and to negotiate on the design problems, and to compromise for having optimized solutions during the design process. However, each design actor usually concerns only in his/her own tasks and does not perceive others'. [Chapa Kasusky 1997] realizes this problem, and then implements the concept of *multi-actor* and *multi-view*. The objective of these concepts is to permit design actors, experts, or contributors from different disciplines presenting their information, knowledge, and constraints into the collaborative environment.

Multi-actor

The word 'actor' in this context means designers, experts, or contributors who contribute any information of a product to the design process, which also can be called 'design actor'. However, in the context of CoDeMo, there are two types of actor: internal actor and external actor (see Figure 4.3).

The internal actor means a computer application that acts as an actor. During the design process, design actors have to contribute and share a lot of information and knowledge to characterize the product by using their *features* (see 4.2.1). Each feature definition includes an implicit reference and complements of specific trade's viewpoint. The internal actor is then developed to associate the initial features to the corresponding implicit form features. The tasks of the internal actor here are to keep the coherence between constraints, so to execute the tasks of system, e.g. data propagation, data translation, constraint propagation [Roucoules 1999], substitution [Radulescu 2005], etc.

The external actor means a real 'design actor', a user, or an expert who contributes information, knowledge, and constraint of the product via the GUI of CoDeMo. To accomplish the multi-actor concept, CoDeMo is implemented to create a virtual meeting room that brings together the design actors to perform the design activities.

Multi-view

To realize the multi-actor concept, [Chapa Kasusky 1997] implemented the multi-view concept that permits the design actors to contribute relevant information and to present the product in their mind by their own view. The multi-view concept takes account that design actors does not concern with the same objective. Each design actor may have different view to decompose the same product or component, depending on his/her interest. One actor defines assembly solutions, another one concerns about mechanical testing while other one might concern the manufacturing process. Therefore, the multi-view concept is required when we need to consider through the product life cycle. This concept creates an own view for design actors in each domain to characterize specific information to the product.

There are two types of view: *trade view*, and *common view*. A trade view is used to represent the product of one's interest. It allows the design actors to describe the product with their specific description. Each design actor could have an own view in order to create new data, and s/he can also modify, edit, or delete the existent information. The design actors use this information to evaluate the design, and the results of evaluation will create new information to the product. A common view, as name defined, is a view that every design actors, who connected to the system, can access and the information stored in this view can be seen. There are, for now, two common views: *frame view* and *geometric view*. The frame view stores the information relative to the functional surface of the product with its characteristics such as roughness, tolerances, etc. The geometric view stores the geometric data and is finally the results of the integration of the trade views.

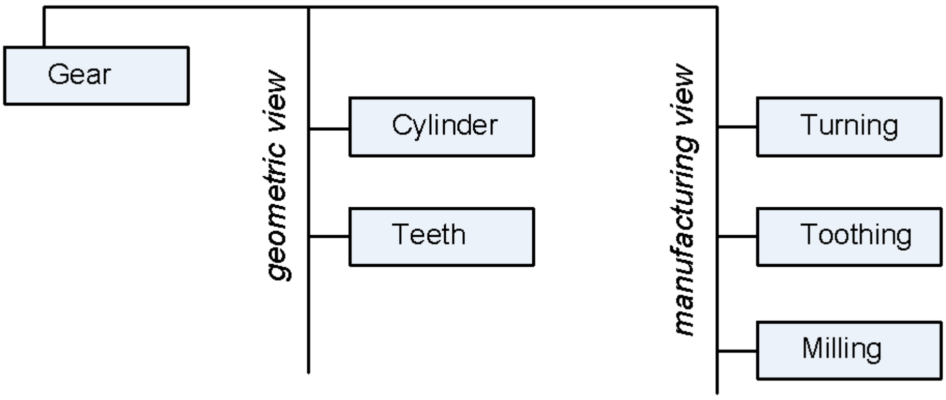


Figure 4.6 Multi-view representation

Multi-representation

In the design process, design team usually deals with numerous of information. The design system must realize this problem. As presented that the product model is structured by the data model. To support the concept of multi-actor and multi-view, the multi-representation is required. It facilitates the design actors to represent their information, knowledge, and constraints by giving them a basic representation as shown in Figure 4.2, a functional representation and a textual representation. Furthermore, it envisions the product model by representing in 3D graphical representation as shown in Figure 4.7.

These three concepts constitute the infrastructure of the integrative environment. This section just introduces moderately the multidisciplinary concept. However, more details and descriptions can be found in [Chapa Kasusky 1997], [Roucoules 1999], and [Roucoules and Tichkiewitch 2000]. To illustrate the multidisciplinary concept, the interaction between design actors will be presented in the next chapter and the demonstration will be presented in Chapter 7.

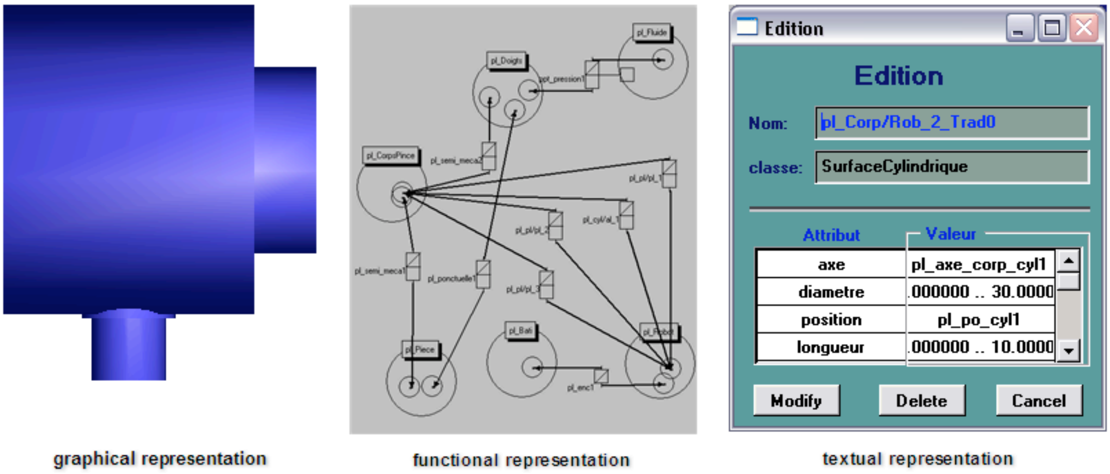


Figure 4.7 Multi-representation

4.3.7 The concept of decomposition

In the Axiomatic Design, to satisfy the functional requirements, the design must be decomposed in different hierarchical levels to characterize the characteristics of the product. A product can be decomposed in different levels depending on the context, e.g. the physical decomposition, the functional base decomposition, the decomposition bases on the activity [Radulescu 2005]. As well as a product, a component can be decomposed into different hierarchical levels of abstraction as shown by example in Figure 4.8. The highest level of decomposition gives an overview of system while the lower level(s) enhances the design actors understanding the system or how the mechanism is made. It also gives possibilities to specify in the least details of parts. In addition, components can be associated with links to define characteristics themselves by a relation as shown in Figure 4.2.

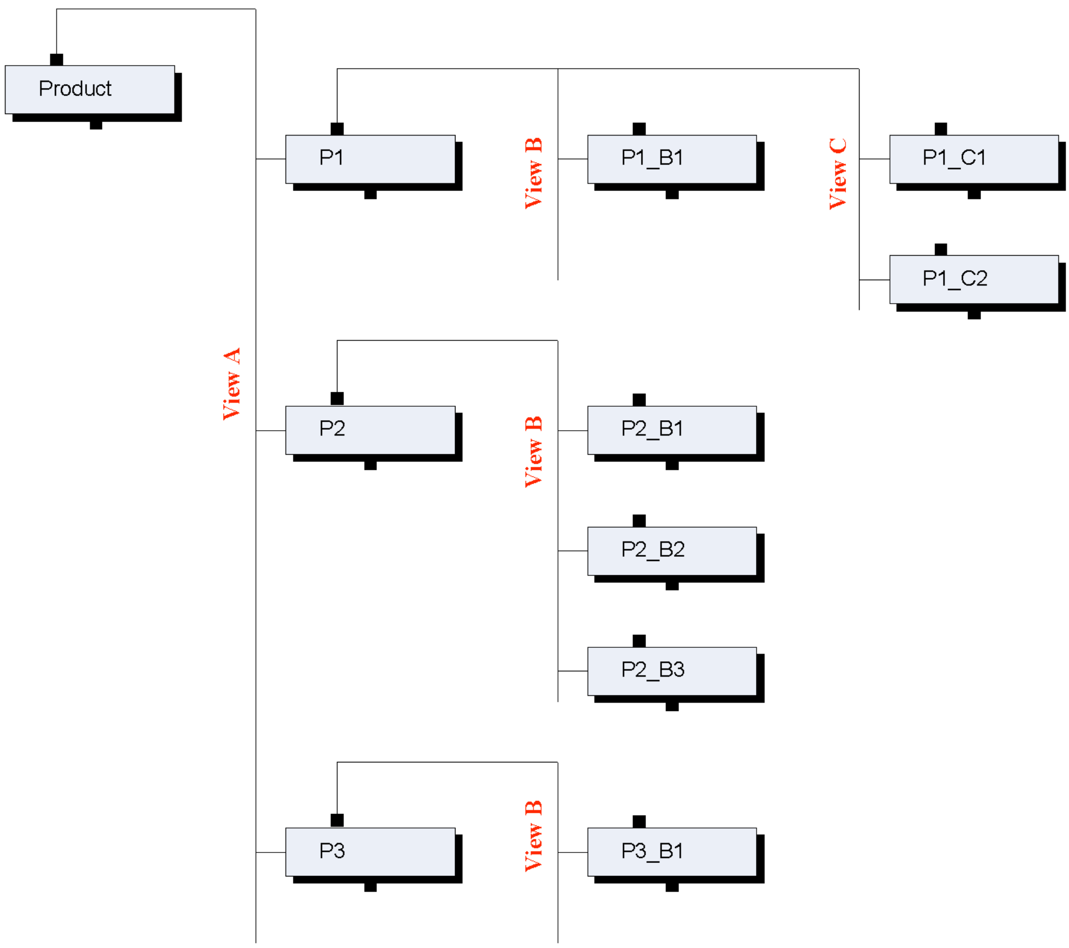


Figure 4.8 Decomposition of a component

4.3.8 Knowledge management method

Knowledge management method is used to capture the knowledge in the design process and to embody that knowledge [Tichkiewitch et al 2006]. During the design process, the design system must handle two important tasks, i.e. one is how to notify the design actors to perceive the shared knowledge and the given constraints of each other, and the other is how to make the design actors comprehend the shared information of each other.

<u>Data propagation</u>

We know well that the imaginary complexity arises due to the unknown and/or the ignorance of the designers. However, in the design system, an uncoupled or decoupled design could be a coupled design due to the absent of notification process. To notify the design actors to perceive the design matrices – the established DPs which mean the information and the constraints created by others, the system must transmit such information to the shared database and only the relevant information will be dispatched to the relevant persons at the right time.

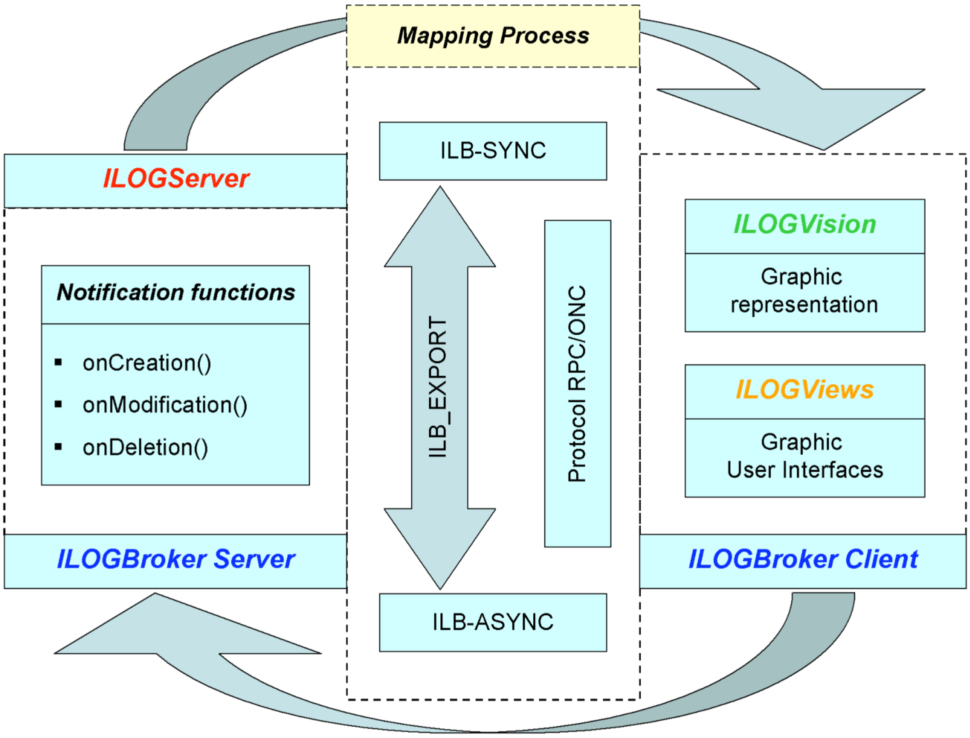


Figure 4.9 Architecture of the data propagation with ILOG libraries

CoDeMo employs the internal actor to occupy this task. This *data propagation* guarantees that the design actors are always up to date and can reach any data and/or constraints in the shared database. Base on the client-server system, when any data is created by an actor, the internal actor will propagate automatically the data to the design actors, who connected to the system and concern such data. The multi-representation permits the design actor to create data via graphic user interface (GUI). Indeed the data is created in the server process (internal actor) not the client process (external actors – design actors). Nevertheless, not only the creation of data but modification, deletion, or any action must be propagated to the server and can be seen to the clients. This *notification function* is developed by using ILOG libraries [ILOG]; *ILOGBroker* libraries supply the Remote Procedure Call (RPC) connections between the server and the clients while *ILOGServer* libraries are used to propagate all actions from the server to all clients and from clients to the server. These two libraries create a *mapping process* that maps the information between the design actors and the shared database and creates also the *notification function* to notify the design actors to perceive the created/modified information. In addition, ILOGViews and ILOGVision libraries provide the GUIs and the multi-representation for the design actors. Figure 4.9 shows the architecture of notification function and CoDeMo. The using ILOG libraries (ILOGServer and ILOGBroker) have been presented in [Roucoules and Tichkiewitch 2000]. This method creates a collaborative environment and enhances the system to be a synchronous system.

Data translation

During the design process, the design actors have to characterize the product as much as they have in their mind into the shared database. We know well that the product data is constituted of numerous information and knowledge. Storing such information/knowledge as descriptive explanations are not easy to manage and difficult to share. To be concise, *feature* is then used to present such information and knowledge by using semantics and attributes. The *product model* is also proposed to associate the knowledge *model* (features) to the *data model* (components, links, and relations), as presented in (4.2).

In the design phase, the design actors use a lot of features to describe the product with their points of view. Each feature is defined by a name and is associated to the trade(s) that concerned. Often a feature definition includes a geometrical implicit reference and complements. These features are concise and facilitate the design actors to characterize the product with their own vocabulary. On the other hand, the design actors may not clearly understand of what one wants to communicate by his/her features due to the vocabulary.

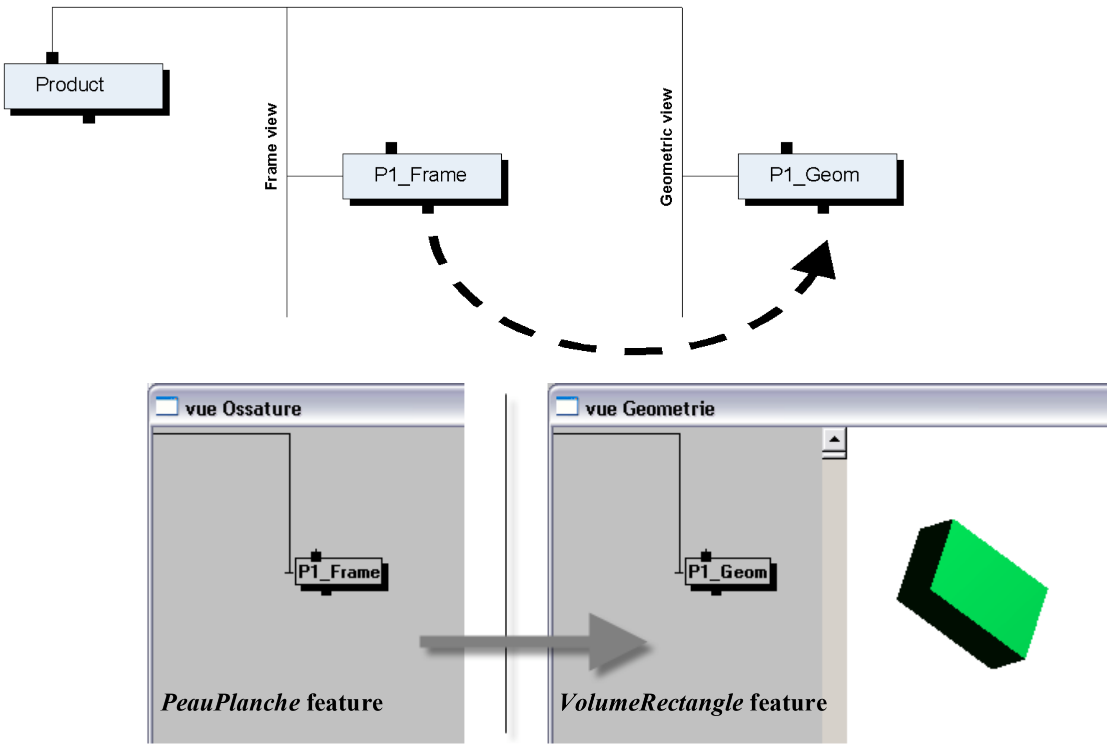


Figure 4.10 Feature translation

To facilitate the design actors, CoDeMo employs the internal actor to manage this task. The *data translation* method is developed to associate with one's features and then interpret those features to another actor(s) who concern into a comprehensible form. For example, *PeauPlanche*[12] is a feature, in the frame view, that represents a function of rectangular surface with its functional characteristics – tolerances. This feature is engaged to be translated as a *VolumeRectangle* feature, in the geometric view, that represents the nominal dimension of functional surface with its geometrical characteristics – length, width, thickness. This means any component in the frame view that associated to the *PeauPlanche* feature has to create another component in the geometric view with the *VolumeRectangle* feature as shown by example in Figure 4.10.

This example of data translation is a general and can be used for any feature. It translates some of *descriptive features* but not all attributes values of characteristics of the features, neither constraint features (relations). However, the objective of the translation method is not to automatically translate one's features to other trade views but to facilitate the design actors to achieve those heavy tasks.

Gathering the information/constraints from every aspect is a complicated task. Therefore, the system should allow the design actors participating in the data translation process. To support this integrated manner, we should create a common space that enhances the design actors to design their features. [Roucoules 1999] introduces a neutral file named '*QTrans*' that is created to assist the internal actor in the translation process. This neutral file is used to store modules of knowledge in a specific grammar. It permits the design actors to acquire the relevant information/features in the modules of knowledge. In addition, it also permits them to add, to edit, and to correct the existing knowledge. The specific grammar of this file is associated to the data translation method, Table 4.1 shows by example a module of knowledge in the *QTrans* file. The features in this *QTrans* file permits the internal actor creating dynamically component, links, and relations as described in the module of knowledge.

Table 4.1 Extract of knowledge model from *QTrans* file

Component_Name
  PeauArbre Ossature name

Traduction
  Link name discipline axe_peau_name
  Component Cylindre Geometrie name_Trad_0
  Link name_Trad_0 peau_origine axe_geom_name
  Relation axe_peau_name name axe_geom_name name_Trad_0 Identite
  name_identite

@

[12] *PeauPlanche* in this context is a French statement of a feature which means *surface of plate*

The result of this *QTrans* can be explained as: when one actor, who concerns, creates a component '*P1*' in *Ossature* view[13], the knowledge model that corresponds to '*P1*' will be created i.e. the component '*P1_Trad_0*' in Geometrie view [14], two links '*axe_peau_P1*' and '*axe_geom_P1*' that associate to those two components and are associated by the relation '*P1_identite*' as show in Figure 4.11. Indeed, the concept of the translation method is not to permit one design actor gathers together constraints and viewpoints of the others but to permit each design actor to contribute information and knowledge to the product with his/her own language.

Note that in the initial design phase, the internal actor also facilitates the design team to recognize the product structure. As soon as the conceptual product model has been brought into the system by the technologist, the internal actor creates automatically features with semantics (e.g. rectangle, cylinder, etc.) and its characteristics (e.g. diameter, length, width, thickness, etc.) followed the default values from the geometrical model of the product (note that these default values would be adjusted during the design process by the design actors). It also establishes some *constraint features* to the parts (e.g. perpendicular, parallel, symmetry, etc.). This initial information enhances the design actors to recognize the product structure – which part contacts with which part in which surface, which parts are parallel to each other, which parts are symmetry, etc.

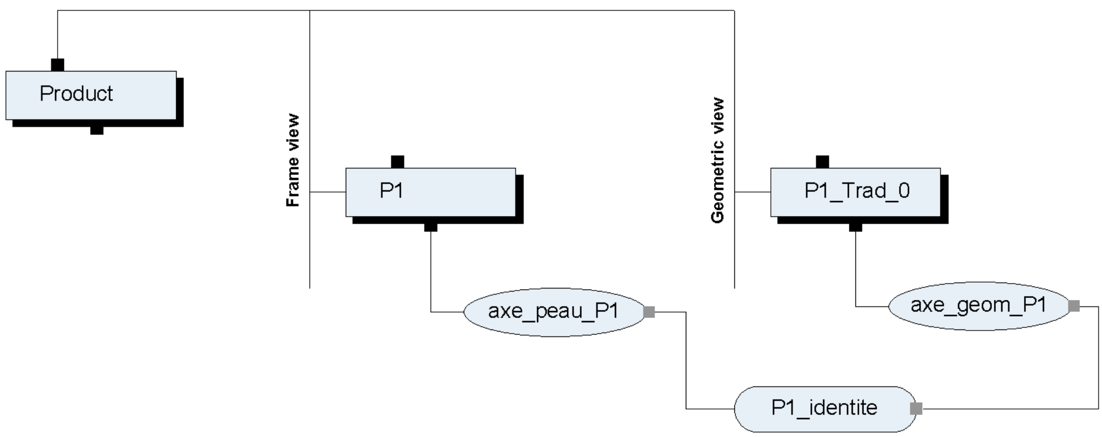


Figure 4.11 Example of knowledge model translation

Furthermore, the internal actor also outputs neutral files that the design actor(s), who concerns, can use (with other specific tools) to evaluate the design in their own view. For example, the forger uses neutral files, translated by the internal actor, as information to optimize the design of stamped parts by using the specific tool,

13 *Ossature* view means *Frame* view

14 *Geometrie* view means *Geometric* view

COPEST [Boujut and Tichkiewitch 1995]. Table 4.2 shows a part of information of a neutral file for using with COPEST.

Table 4.2 Example of a neutral file used in COPEST

```
MACHINE
PRESSE_2

TYPE_DE_MATERIAU
ACIER_NA

TYPE_DE_TRAVAIL
CHAUD

**************************************************
   1.00000000    1.00000000    6.00000000   -186.00000000    0.00000000    0.00000000
200.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.0000000E+00
   0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000

**************************************************
   2.00000000    2.00000000    3.00000000   -18.01004028    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.0000000E+00
   0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000

**************************************************
3.00000000    3.00000000    -3.00000000    -18.01004028    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.0000000E+00
   0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000

**************************************************
   4.00000000    4.00000000    6.00000000   -186.00000000   -10.00001621    0.00000000   -
186.00000000    10.00001621    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.2456461E+25
   0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000

**************************************************
   5.00000000    4.00000000    6.00000000   -186.00000000    10.00001621    0.00000000   -
167.00000000    10.00001526    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.2456461E+25
   0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000    0.00000000    0.00000000    0.00000000
0.00000000    0.00000000    0.00000000

**************************************************
```

Substitution method

[Radulescu 2005] introduced the substitution method that supports the design actors to substitute a relation or a constraint feature with a solution, such as an assembly solution. Based on the concept of data translation, the substitution method employs the "*QTrans*" file to store some modules of knowledge of available solutions that are used to replace the existing feature constraints. The replacement may consist of a set of components, links, and relation(s) or only links, and relation(s) as shown in Figure 4.12. Contrary to the decomposition, the substitution does not change the level of abstraction.

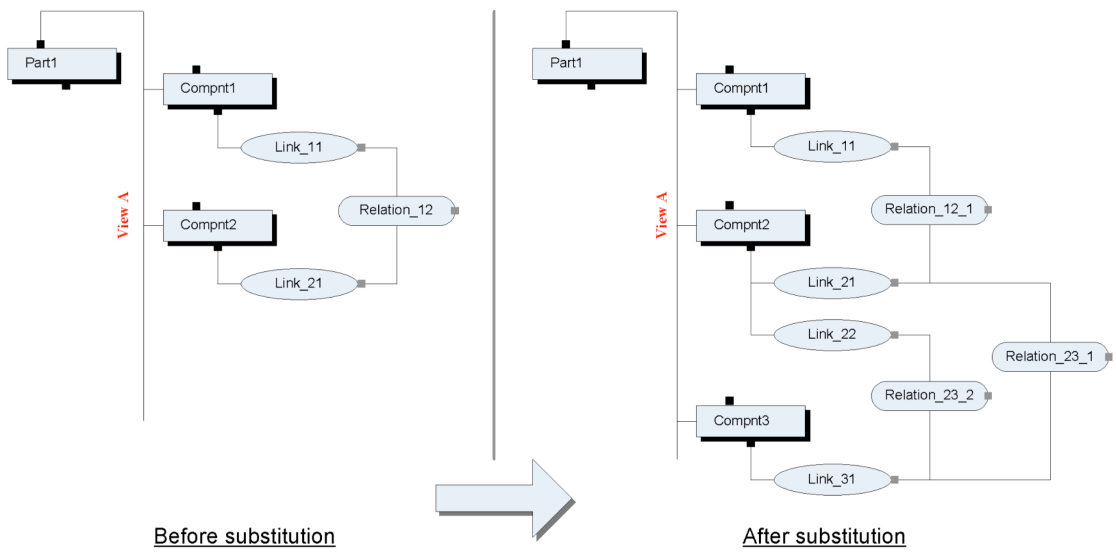


Figure 4.12 Example of substitution method

4.4 Summary

This chapter has examined the studies conceived by the integrated design team of G-SCOP laboratory. It can be summarized that the integrated design consists of *methods*, *models*, *tool for integration*, and *specific trade applications*. The integrated design concept and the collaborative environment allow the design actors to work in collaboration, to access the shared information, and also to add, edit, modify, or delete the information. The model for integration is used to support the knowledge integration. The concept of product model: data model structures the skeleton of the product, and knowledge model capitalizes knowledge of different trades. We use the multidisciplinary concept to organize the design activities. In addition, the concept "worlds of design" and intermediary objects allows actors to communicate to each other although they are in different trade. The concept of decomposition and multi-view representation allow actors to project their knowledge and characterize the details of a product with their own point of view. The knowledge management method facilitates the design actor to be up-to-date. It notifies the design team to perceive the established constraints and information. The data propagation method is provided for mapping the information between the design actors and the shared database. The data translation method interprets the information from one actor to another actor(s) in comprehensible form and also allows the design actors to solve the contradictions together. And the substitution method facilitates the design actors to replace a constraint feature with a solution.

These methods facilitate the design actors to evaluate the design, to communicate to each other, discuss on the design problems, negotiate, and compromise as a socio-workgroup. This chapter has presented the methods for integration and models for integration while the tool for integration (design modeller) and the specific trade application will be presented in Chapter 7 and 8 respectively.

Chapter 5

Acquisition of Knowledge Model in Trade Views

We have introduced the concept of product model, which is comprised of knowledge model and data model, in the fourth chapter. This chapter aims at presenting the constitution of knowledge model of the design actors. It presents how the design actors introduce knowledge model: factual knowledge and temporal knowledge, how to manipulate such knowledge into the product, and how the design team shares and exchanges their information during the design process.

5.1 Introduction

Due to the shortcoming of increasing storage and difficulty of transportation, furniture today has been developed to overcome such problems by designing products as a furniture kit which is called knock-down furniture. This sort of furniture is supplied as a kit of flat parts and fasteners, and then is packed into a carton. It allows customers to assemble the product by themselves. In addition, to make furniture to be more affordable and to give a reasonable price, furniture made of particleboard and fiberboard has been introduced. Formerly, it was not quite cheap and its design lacked strength and stability particularly under heavy loads. The furniture was identified as feeble and unreliable as well as its fasteners. Screws or nuts and bolts were used to fasten the components but it was difficult to quick disassemble. Furthermore, repeated assembly and disassembly may strip the fasteners and the parts, which consequently result in their failure. However, the current fastening system has been developed to overcome these disadvantages of the prior known system by inventing diversity of fasteners that provide a reliable fastening system and the user can quickly assemble and disassemble the furniture. Since the manufacturing technology has been more developed, particleboard and fiberboard become cheaper and have better in quality. Various particleboards and fiberboards have been created to support different kinds of work. These advantages permit producers to select the most appropriate materials to their products. The manufacturing system of this sort of furniture has been as well developed to support the era of mass production. It facilitates manufacturers to manage the production planning, process planning, and manufacturing cost.

We know well that the "*over-the-wall*" syndrome causes serious problems and numerous iterative interactions in the design process. We are then obliged to gain knowledge and constraints of different phases as much as possible and bring into the early stage of the design process. The design actors have to manipulate such knowledge and constraints by using data translation and data propagation method and by taking advantages of the concept of product model, and also by associating the knowledge model to the data model in order to store information into the shared database and to present it through GUIs. We present in this chapter how to constitute the knowledge model – features and production rules that present know-how of the design actors.

5.2 Features and production rules

We have proposed in the integrated design system that the design actors must participate to the common tasks and introduce their constraints to the design as soon as they perceive. We have briefly introduced in Chapter 4 that knowledge model is characterized into two categories i.e. *features* (factual knowledge) and *production rules* (temporal knowledge). This knowledge model permits us to contribute information and to define constraints which are associated to data model. These two types of knowledge have to be manipulated to construct the product model.

5.2.1 Features

A feature is a semantic object manipulated by a design actor and is used to define the product. A feature is given a name by the user who creates it. Thus, the meaning of a feature is different depending on the context in which it is created. A feature is described by characteristics and behaviors. For example, the feature '*Tourillon*'[15] is a sort of fastener in the assembly view. It is used as a guide and to fix two parts together. The characteristics or *descriptive features* of *Tourillon* can be described as diameter, length, type of material, etc. Values of characteristics of a feature are normally defined at the end of the design process. Nevertheless, they may have some admissible or initiate values by default. Note that for each use of any feature, the system creates an *instance* of such feature. At the end of the design process, each value of characteristics of each created instance must be known. If there is any unknown value, it means that the design process has not finished yet. A

[15] *Tourillon* is a French word that means a dowel

behavior is a method that links two or more characteristics of feature(s), which can be from one or different features. A behavior of a feature can be defined as a *constraint* of characteristics. A role of a feature can be either a component or a relation while a role of characteristic is a link.

A feature may be considered as an element of knowledge relative to a design actor. Each trade engaged in a design view has its own library feature. Some features represent the knowledge of a specific trade and are only used by a specific design actor in the trade. Some features are recognized and used by several design actors from different trades. Such features have significance to those actors and are considered as multi-context features. They are treated as a communication object by the design actors in order to discuss, to negotiate, and/or to compromise the design. To classify the characteristics of features, [Gaucheron 2000] proposes the taxonomy of features into three significations as following.

- Vernacular feature – represents the knowledge relative to a specific trade and is always available to the specific design actor who created it. For example, the '*AssemPlanche*'[16] feature is concerned by the assembler in the assembly view.
- Vehicular feature – can be recognized and used by several design actors of different trades, who interest on the same information. By example, the '*Tourillon*' feature is concerned by the assembler but it is affected by the thickness of the parts that defined in mechanical view. It affects also the manufacturing process planning of the manufacturer. This feature is considered as communication object to support the coordination where negotiation and compromise between design actors are needed [Noël and Tichkiewitch 2004].
- Universal feature – can be recognized for everyone and used in common view; usually is a part of collective views. This feature facilitates the notion of integrated design and allows design actors to negotiate.

To create and to use such features relative to one's trade, that trade must have an access to a *feature based engine* (see Figure 7.1 in Chapter 7).

5.2.2 Production rules

Production rules are elements of knowledge used in an expert system, which may be found in different books on artificial intelligence. A production rule is an element of an activity model, which begins with a premise (If A) and ends with a conclusion (Then B) [Brissaud and Tichkiewitch 2000]. It creates a notion of temporality and it is used in problem-solving process. The premise A may be a fact that is concerned by an instance of a feature or one or more values of characteristics.

[16] AssemPlanche means to a plate for assembly

The conclusion B may define one or more values of characteristics of existing instances of feature, or may create new instances of feature as new elements of the product or launch a specific application. The temporal knowledge is used to present a strategy of design actors.

In the design process, one cannot solve the design problem by oneself because there is obligatory an interaction between the actors. During the design process, if the assembler has chosen an assembly solution, there will be automatically a consequence in the manufacturing view. Likewise, the choices of the mechanician also have a consequence to the assembly view. So, we have to ask these actors to work together and to find out the interactions between their views. In the study of Gaucheron, he presented that the actors from different disciplines have to prepare and discuss on the common problems before they begin the design process. The interaction between the actors can be expressed by dialogues or discussions in the form of production rules. Let's see an example of fastening the two parts by using a feature '*Tourillon*' as represented in Figure 5.1 (a) and (b).

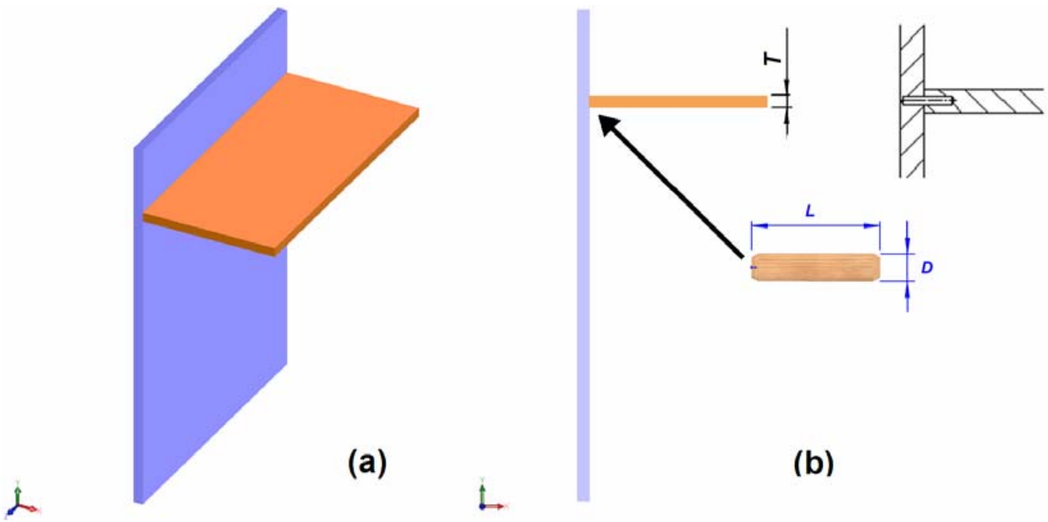


Figure 5.1 Example of using a dowel as an assembly solution

At this time, the design actors may create some production rules of feature '*Tourillon*' as represented in Table 5.1. In order to use a feature '*Tourillon*', we must take into account the length and the diameter of the dowel, and also the thickness of vertical part and horizontal part. This example of production rules contains *coherent* information between trade views. The assembler uses *dowels* as an assembly solution for these two parts. The mechanician defines the thickness of the parts that is relative to the characteristics of the dowels. The characteristics of the dowel influence the characteristics of manufacturing process. In order to create and to employ a production rule, the design actors must have an access to an *inference engine* (see Figure 7.1 in Chapter 7). In order to share this knowledge, we must introduce it into the integrated design system by transforming into the "*QTrans*" file as presented in Chapter 4.

Table 5.1 Temporal knowledge of feature Tourillon

If a dowel is applied to fix a pair of parts
Then those two parts must be drilled
If the thickness of the horizontal part is *T* mm
Then the diameter of the dowel is not more than *T/2* mm
If the diameter of the dowel is *D* mm
Then those two parts must be drilled with diameter *D* mm
If the length of the dowel is *L* mm
Then the horizontal part is drilled *2L/3* mm while the vertical part is drilled *L/3* mm

5.3 Constitution of knowledge in assembly trade view

In the assembly view, the objective of the assembler is to examine the possibilities of assembly solution and then to choose the most appropriate solution for the parts. The system creates a library of assembly solutions in the assembly view. This library contains features for assembly solutions and its characteristics as represented for example in Table 5.2. From these examples, we can see that a feature of assembly can be a material component (such as dowel, screw, etc.), which is used as a supplementary part, or an operation of manufacturing (such grooving), which does not need any supplementary part for assembly.

Table 5.2 Examples of descriptive features for assembly solution

Features and characteristics	
Dowel Type (Strand, Groove) Diameter Length Material (Wood, Metal) Maximum load	 
Grooving Type (Through, Distant) Width Depth Length	

The assembler must choose an assembly solution (a design parameter, in the complexity theory) for each assembly problem detected in the technological view. However, s/he could not define values of the DP until s/he has known the dimension of the parts which normally is defined by the mechanician. Yet, the assembler must concern the properties of the chosen fasteners and also the load that the parts must be supported. The assembler may refer to the load given by the referred standard. These constraints affect the choice of assembly solution. [Soltis 1999] presents by example a tapping screw that is commonly used to fasten particleboard where withdrawal strength is important. The assembler must be careful when tighten a screw into the particleboard to avoid stripping the threads. The maximum torque that can be applied to a screw before the threads in the particleboard would be stripped is given by

$$T = 3.16 + 0.0096X \tag{5.1}$$

where T is torque (N-m)

X is density of the particleboard

This equation is for 8-gauge screws with a depth of penetration of 15.9 mm. (5/8 inch). The maximum torque is fairly constant for lead holes of 0 to 90% of the root diameter of the screw. The ultimate withdrawal loads P (N) of screws from board can be predicted by

$$P = KD^{1/2}(L - \frac{D}{3})^{5/4}G^{2} \tag{5.2}$$

where D is shank diameter of the screw (mm.)
L is depth of embedment of the threaded portion of the screw (mm.)
G is specific gravity of the particleboard
K is 41.1 for withdrawal from the face or is 31.8 for withdrawal from the edge of the particleboard

This equation is applied when the setting torque is between 60% ~ 90% of T. A modest tightening of screws in many cases provides an effective compromise between optimizing withdrawal resistance and stripping threads. Equation (5.2) can also predict the withdrawal of screws from fiberboard with K = 57.3, for the face and K = 44.3 for the edge of the board.

It can be concluded that the constraints in the mechanical view and the choices of assembly solutions may constrain each other. Yet, they also influence the definition of features and their characteristics of manufacturing process in the manufacturing view. For example, the vertical part in Figure 5.1 is a shelf part of a desk and is described, in the assembly view, with an assembly constraint *Tourillon* and its characteristics as shown in Table 5.1.

In order to allow production rules to be used with CoDeMo, the common file *QTrans* is developed to stores modules of knowledge. This *QTrans* file is developed to assist the internal actor in the system to translate the modules of knowledge from different actors. We present here, for example, a module of knowledge in the *QTrans* file which is developed for sharing knowledge between the assembly view and the manufacturing view. It contains a set of rules with some features and their characteristics. To apply such production rules, the GUI in the assembly view proposes the assembler a set of admissible solution to substitute the assembly relation. The chosen solution will replace the relation and create the corresponding information stored in the QTrans file as presented for example in Table 5.3. The assembler can later define values of the characteristics.

Table 5.3 Production rules for feature Tourillon in QTrans file

```
Component_Name
        Tourillon Assem name
Traduction
        Component Percer Usinage name_1_USI
        Component Percer Usinage name_2_USI
        Link name diametre name_diametre
        Link name longueur name_longueur
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Link name_2_USI diametre name_2_USI_diametre
        Link name_2_USI epaisseur name_2_USI_epaisseur
        Relation name_diametre name name_1_USI_diametre name_1_USI relation_name_1
        Relation name_longueur name name_1_USI_epaisseur name_1_USI relation_name_2
        Relation name_diametre name name_2_USI_diametre name_2_USI relation_name_3
        Relation name_longueur name name_2_USI_epaisseur name_2_USI relation_name_4
@
```

A *QTrans* file is a set of proposition separated by a symbol “@”. One proposition comprises keywords that construct the structure of *QTrans* file, i.e., ‘*Component_Name*’, ‘*Traduction*’, ‘*Component*’, ‘*Link*’, and ‘*Relation*’. We can describe such key words as following:

- *Component_Name* initiates a proposition which is followed by a name of feature ‘*Tourillon*’, a corresponding view ‘*Assem*’[17], and a name of instance of feature which is defined by a user.

[17] *Assem* view means to *Assembly* view

- *Traduction* implies operations of translation process. It is followed by following key words:
- *Component* implies a creation of a component object. For example, the first line of *Traduction* creates an instance of feature '*Percer*'[18] in the view '*Usinage*'[19] with name of the instance and is followed by "_" and number of component and "_USI".
- *Link* implies a creation of a link object. For example, the third line of *Traduction* creates a link for the instance which is associated to a characteristic '*diametre*'[20] with the name of the instance and the name of the characteristic.
- *Relation* implies a creation of a relation object. It is followed by a name of a link of the instance, the name of the instance, a name of a link of a new instance, the name of a new stance, and a name of a relation. For example, the ninth line of *Traduction* create a relation for the instance and the new instance number 1 by associating their link with a name "relation_" followed by the name of the instance, "_", and the number of the instance.

Nevertheless, the characteristics of features need to be evaluated. To evaluate values of the characteristics such as the diameter and the length of dowel, diameter and depth of hole, they need to know first the thickness of parts. Thus, the results from the mechanical view are required. We will present continually the production rules of this example in the next section. In addition, other examples of descriptive features for assembly solution and examples of production rules used in trade views and common views can be found in Annex I.

5.4 Constitution of knowledge in mechanical trade view

In the mechanical view, the task is mainly to test the deflection and the deformation of the parts. In this section, we examine the structural analysis equations, which are the basis for beam and column design, to determine the deformation of the part. The results of the test are dedicated to the choice of material type and thickness. The deformation equations are represented as functions of applied loads, module of elasticity, module of rigidity, and parts' dimension. They are applied to determine the minimum required thickness (cross-sectional dimension) of the parts to meet the deformation limitations imposed by the quality view. Consideration must be given to

[18] *Percer* is a French word that means to drill, in this context we mean to the 'drilling' operation

[19] *Usinage* view means to *Manufacturing* view

[20] *diametre* is a French word that means *diameter*

variability in material properties and uncertainties in applied loads to control reliability of the design.

5.4.1 Compression load

The deformation of an axial load produces a change of length which is given by

$$\delta = \frac{PL}{AE} \tag{5.3}$$

where δ is change of length
P is axial force
L is length of beam (part)
A is cross-sectional area
E is modulus of elasticity

5.4.2 Bending and shear

In this study, we concern the deflection of straight beams that are elastically stressed and have a constant cross section throughout their length. The deflection of straight beam is given by

$$\delta = \frac{k_b W L^3}{EI} + \frac{k_s W L}{GA'} \tag{5.4}$$

where δ is deflection
k_b and k_s are constants dependent upon beam loading, support conditions, and location of point whose deflection is to be calculated
W is total beam load acting perpendicular to beam neutral axis
L is beam span
E is modulus of elasticity
I is moment inertia of beam
A' is modified beam area
G is shear modulus of beam

The first term on the right side of Equation (5.4) gives the bending deflection and the second term gives the shear deflection. The values of k_b and k_s are represented in several cases of loading and support, given in Table 5.4.

The moment inertia I of a beam is given by

$$I = \frac{bh^3}{12} \quad \text{for rectangular cross section beam} \tag{5.5}$$

$$I = \frac{\pi d^3}{64} \quad \text{for circular cross section beam} \tag{5.6}$$

where b is width of beam
h is depth of beam
d is diameter of beam

The modified area A' is given by

$$A' = \frac{5}{6}bh \quad \text{for rectangular cross section beam} \quad (5.7)$$

$$A' = \frac{9}{40}\pi d^2 \quad \text{for circular cross section beam} \quad (5.8)$$

Table 5.4 Values of k_b and k_s for several beam loadings [Soltis 1999]

Loading	***Beam ends***	***Deflection at***	k_b	k_s
Uniformly distributed	Both simply supported	Midspan	5/384	1/8
	Both clamped	Midspan	1/384	1/8
Concentrated at midspan	Both simply supported	Midspan	1/48	1/4
	Both clamped	Midspan	1/192	1/4
Concentrated at outer quarter span points	Both simply supported	Midspan	11/768	1/8
	Both clamped	Load point	1/96	1/8
Uniformly distributed	Cantilever, one free, one clamped	Free end	1/8	1/2
Concentrated at free end	Cantilever, one free, one clamped	Free end	1/3	1

Table 5.5 Examples of materials type used in the mechanical view

Name	***Physical mechanical properties***			***Screw-holding***	
	Modulus of Elasticity *(N/mm^2)*	***Modulus of Rupture*** *(N/mm^2)*	***Internal bond*** *(N/mm^2)*	***Face*** *(N)*	***Edge*** *(N)*
H-1	16.5	2400	0.90	1800	1325
H-2	20.5	2400	0.90	1900	1550
110	14.0	1400	0.30	780	670
120	14.0	1400	0.50	875	775

The mechanician disposes different materials which have a specific name and values of characteristics such as module of elasticity, module of rigidity, and other mechanical properties. It is depending on the standard and producers that we refer to. Table 5.5 shows, by example, some materials and its mechanical properties. Other examples of materials type that used in the library of the mechanical view can be found in Annex I.

The result of deflection depends on one hand, the given load regarding to the referred standard, which the assembler can not control; on the other hand, the type of material and the thickness of the parts. The objective of the mechanical view is that to define the most appropriate materials and thickness for the product. The choices in the mechanical view are relative to assembly solutions and characteristics of the manufacturing process, as presented before in Figure 5.1 and in Table 5.1.

If the thickness of the horizontal part is *T* mm
Then the diameter of the dowel is not more than *T/2* mm

After the assembler has evaluated the design, s/he may output the results in a form of text file (will be presented in Chapter 7). As soon as the type of material and the thickness of the parts have defined, the actors who are concerned to such information will consequently take into account and continue their evaluation. From the problem presented in Table 5.3, since the mechanician has defined the thickness of parts, we continue to present an example of production rules between the mechanical view and the assembly view, as presented in Table 5.6.

Table 5.6 Production rules for feature Tourillon between mechanical and assembly view

```
Component_Attribute
    PlancheMeca Meca name_MECH
    Tourillon Assem name_ASM
Traduction
    Attribute name_MECH materiau Char materiau_planche
    Attribute name_MECH epaisseur Float epaisseur_planche
    Attribute name_ASM epaisseur Float epaisseur_planche
    Link name_MECH epaisseur name_MECH_epaisseur
    Link name_ASM epaisseur name_ASM_epaisseur
    Relation name_MECH_epaisseur name_MECH name_ASM_epaisseur name_ASM
    relation_MECH_1
@
```

In order to provide relative values of characteristic of features between trade views, we have developed the structure of knowledge module in *QTrans* file. The keywords, ‘*Component_Attribute*’ and ‘*Attribute*’, have been added into the structure. We can describe this new structure of knowledge module as following:

- *Component_ Attribute* initiates a proposition which is followed by a group of coherent instances. In this example, there are two instances that are concerned with this proposition. One line presents one instance which is comprised of a name of feature, a corresponding view, and a name of instance of feature which is defined by a user.

- *Traduction* implies operations of translation process. It is followed by following key words:
- *Attribute* implies a creation of an attribute of a feature. In other words, it is to define value to a characteristic of a feature. For example, the first line of *Traduction* is to define the type of material for the chosen (*name_MECH*) instance in the mechanical view. The characteristic '*materiau*'[21], which has the type of value '*Char*', is defined by the value '*materiau_planche*'. The value of '*materiau_planche*' is retrieved from the text file that stores the results of mechanical evaluation.

While the keywords "*Link*" and "*Relation*" imply a creation of a link object and a relation object, as same as presented in Table 5.3. As we have noticed that the characteristics of features in the assembly view need to be further evaluated as soon as having the results from the mechanical view. As a result, the assembler can define values of such characteristics of features of the chosen assembly solutions. Following the problem in Table 5.3 and Table 5.6, we may continue to present an example of production rules between the assembly view and the manufacturing view, as presented in Table 5.7.

We have append another structure of knowledge module in *QTrans* file in order to provide such coherent values of characteristic of features between these trade views. A keyword '*SubComponent_Name*' and '*SubComponent*' have been added into the structure of knowledge module. We can describe this new structure of knowledge module as following:

- *Component_Name* initiates a proposition for defining values of characteristics of chosen instance of feature in the corresponding view.
- *SubComponent_Name* implies a group of coherent instances of feature and their characteristics in the corresponding view. In this case, the characteristics of the sub-components in the manufacturing view, which are relative to the chosen component in the assembly view, will be defined.
- *Traduction* implies the beginning of operations of the translation process.
- *Attribute* implies a creation of an attribute of a feature, which has the same structure as described in Table 5.6. In this case, the chosen instance in the view '*Assem*' and the coherent instances in the view '*Usinage*' are defined. For example, the second line of *Traduction* is to define the diameter of fastener '*Tourillon*' for the chosen instance in the assembly view. The characteristic

[21] *materiau* is a French word that means *material*

‘*diametre*’, which has the type of value ‘*Float*’, is defined by the value ‘*diametre_tourillon*’. The value of ‘*diametre_tourillon*’ is retrieved from the text file that stores the results of mechanical evaluation.

Table 5.7 Production rules for characteristics of feature Tourillon in QTrans file

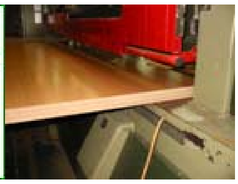
```
Component_Name
    Tourillon Assem name

SubComponent_Name
    SubComponent Percer Usinage name_1_USI
    SubComponent Percer Usinage name_2_USI

Traduction
    Attribute name type Char type_tourillon
    Attribute name diametre Float diametre_tourillon
    Attribute name longueur Float longueur_tourillon
    Attribute name quantity Int qty_tourillon
    Attribute name_1_USI diametre Float diametre1_tourillon
    Attribute name_1_USI epaisseur Float epaisseur1_tourillon
    Attribute name_2_USI diametre Float diametre2_tourillon
    Attribute name_2_USI epaisseur Float epaisseur2_tourillon
@
```

Other examples of production rules used in trade views and common views can be found in Annex I. In order to define characteristics of some features, we may need a specific application to evaluate the design. In this study, we have employed the application “DAPP” that we had developed for using in the manufacturing view, which will be presented in Chapter 7. We present in the next chapter how the design team performs their design tasks by using production rules.

5.5 Constitution of knowledge in manufacturing trade view

To illustrate the industrial area of this sort of furniture, we introduce first the manufacturing processes that are mainly applied to this sort of furniture as shown in Table 5.8. We can describe the manufacturing processes as following:

Table 5.8 Manufacturing process of this sort of furniture

Cutting 	Cutting – is to cut a plate into a desired size
Framing 	Framing – is to structure an assembly-part that is constituted form several small parts and is covered by thin plates. This sort of part is required to decrease the weight and also to economize the cost of materials, and sometimes to satisfy the aesthetic if the thick dimension is required. On the other hand, it may increase the operation cost.
Routing 	Routing – is to shape a plate in a curve-form.
Grooving 	Grooving – is to make a groove to a plate. It may be either a through-grooving or a distant-grooving.
Edge-banding 	Edge banding – is to cover the edges of a plate with some edge bands (made from PVC, ABS material, or melamine). It may be either a straight-banding or a curve-banding.
Drilling 	Drilling – is to drill a plate a hole(s) for fastening with fastener(s).
Finishing 	Finishing – is to tidy up a plate, to attach a gadget(s) to the plate such as a CD support, to label a part number, etc.
Packing 	Packing – is to arrange and to pack all assemble-parts as a product.

Nevertheless, the manufacturer may define the number of manufacturing process more or less depending on the characteristics of the plant and of the product. In the manufacturing view, the manufacturer gathers the information contributed by the other actors to plan the manufacturing process. We present here, by example, a conceptual model of a computer desk named *DS100*, as shown in Figure 5.2. Suppose that it has enough information to be evaluated. The manufacturer may define a process route of manufacturing processes for each part of the product as represented by an *Operation Process Chart* (OPC) in Figure 5.3.

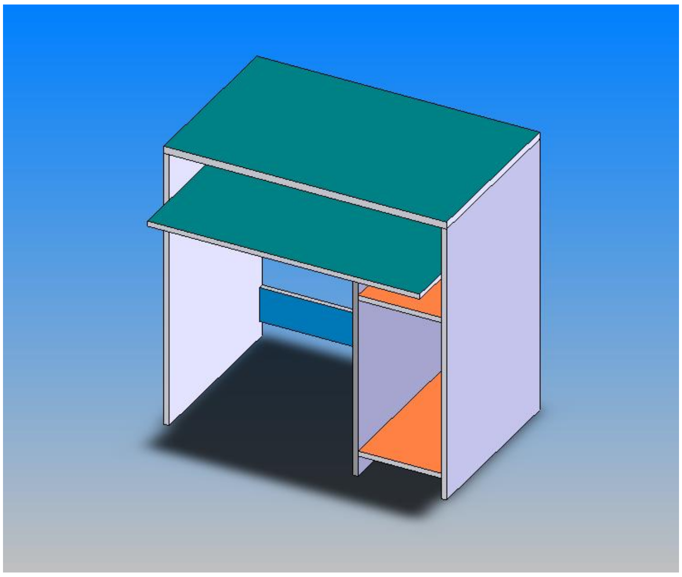


Figure 5.2 Conceptual design of a computer desk

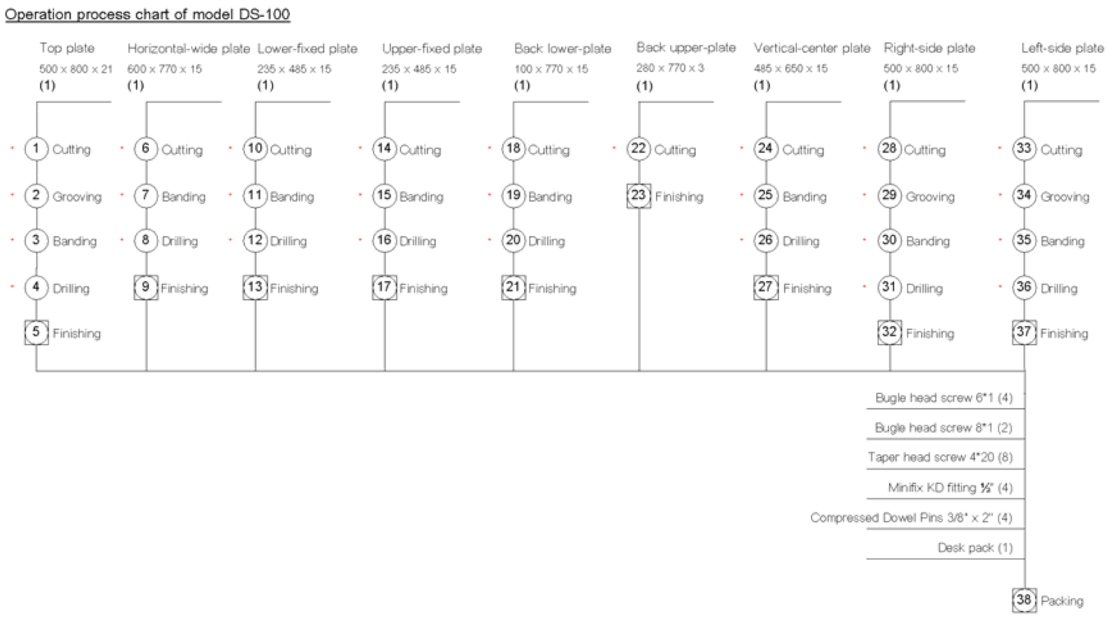


Figure 5.3 Example of operation process chart

This OPC gives an overview of the entire process of operations and inspections of the product. It facilitates the manufacturer to plan the manufacturing process and also to evaluate the design. To capture this knowledge of the manufacturer, we introduce in this study a specific application name DAPP. It manipulates the manufacturing knowledge into the database and facilitates the manufacturer to evaluate the design. This application will be presented in Chapter 7.

5.6 Summary

We have presented in this chapter the constitution of knowledge model in trade views. We apply features and production rules to store knowledge of each actor and manipulate such knowledge into the design process. The assembler chooses an appropriate assembly solution for fastening the parts. The mechanician defines material type and thickness for each part. The contributions from the mechanician enhance the assembler to define characteristics of the assembly solutions (fasteners). In the meantime, the characteristics of assembly solution are relative to manufacturing process of the parts that contribute the manufacturer to plan the manufacturing process. With the preliminary discussion between the design actors, the concept of production rules, and the translation method of modules of knowledge in the *QTrans* file support the design actors to share their knowledge, to exchange information and some constraints, and also to avoid of having some conflicts during the design process.

Chapter 6

Trade Integration for Solving Complex Design

The design process is primarily considered to satisfy a large number of function requirements. This chapter proposes a method for reducing complexity in the design process. It is essential that designer actors or contributors, who intervene at any time in the product life cycle, be presented during the design process in order to introduce their information and constraints. Suh states that the time-independent imaginary complexity can occur when we must satisfy many function requirements at the same time. We propose a solution for solving such complexity.

6.1 Introduction

Due to the globalization, the design and manufacturing of advanced systems requires dozens of engineers and experts, who are usually decentralized in different locations. As design problems today have become more and more complex, people have developed organizations for breaking down the complex problems. Many companies acquire smaller groups in an effort to reduce cost and increase strategic synergies even though the transitions create additional complexity. Unfortunately, this approach sometimes means that the size and scope of each new project increases, so does resulting cost, number of people, lead time of solving problem, and management difficulty. For example, Henry Ford mastered the decomposition method for manufacturing of Ford's Model-T car by breaking down a car into small series of assembly. This method facilitates workers to learn and to perform the tasks easily and quickly. However, individual workers knew only what they were doing with their one perspective. Without the understanding of the whole car, individuals had difficulty to identify problems. Although they found problems, they were unable to fix it by themselves [Stagney 2003]. This led to another problem of complexity.

We have presented four types of complexity in the third chapter. In this chapter, we emphasize the time-independent imaginary complexity, which often arises in the design process. [Suh 2005] defines that time-independent imaginary complexity is uncertainty that arises because of the designer's lack of knowledge and understanding of a specific design itself. Therefore, the time-independent imaginary complexity is considered to be solved. The following sections describe how to reduce the time-independent imaginary complexity by identifying the structure of design matrices, based on the theory of complexity of Axiomatic Design.

6.2 Method for solving complex design

In his keynote, [Suh 2005] presents; "In the future, engineered systems will become more complicated since the number of the functional requirements (FRs) will continue to increase requiring many layers of decomposition, unless fundamental principles for reducing complexity can be devised. Complexity of these systems will depend on our ability to successfully synthesize and operate large systems without making them complex". In fact, the complexity does not always depend on the number of FRs that we have to take into account, if we have chosen the right DPs that satisfy the FRs. Suh defined the imaginary complexity as uncertainty that arises because of the designer's lack of knowledge and understanding of a specific design itself. He described in addition that "When there are many FRs a system must satisfy at the same time, the quality of design in terms of the independence of FRs affects the uncertainty of satisfying the FRs. The uncoupled design is likely to be least coupled. However, the complexity of a decoupled design can be high due to imaginary complexity if we do not understand the system – it is not really complex, but appears to be complex due to our lack of understanding." Although a good design, imaginary uncertainty can exist when we are ignorant of what we have.

As presented previously, knowledge model comprises *features* (factual knowledge) and *production rules* (temporal knowledge). Values of *features* can be affected by the interaction between the design actors regarding to the type of features – vernacular, vehicular, or universal feature – presented in (5.2.1). This interaction is incited implicitly by *production rules*. Design actors use their *production rules* to augment the information of the product into the design process. Then, other actors who concern such information may enhance their knowledge model and may reply or further circulate their information to the team if necessary. This interaction process consequently reveals the concealed information and resolves the imaginary complexity step by step.

Before performing the design process, we postulate in this study two hypotheses as following:

- We suppose that the design problem is an imaginary complexity. In other words, unknown design matrices are triangular.
- To create a collaborative environment, we propose an integrated design system which brings the design team into a virtual meeting room. Each design actor in the team has his/her own knowledge on design problem and may have an access to the existing data of the problem.

In addition, the design actors in the team must have the coincident notion of design, which is called “just need” [Brissaud et al 1997], as following:

- Each actor has to contribute his/her constraints as soon as s/he can. This notion enhances other actors to have further information to evaluate the design and to define the product more precisely.
- Each actor has to contribute the constraints that s/he can prove. To emphasize the previous notion, this notion permits the actors to contribute only the constraints that s/he can prove but not as s/he wants to. The actor must be able to prove that, what he says is necessary to take into account of such constraints.

These notions facilitate the design team to perform the design tasks with less problems and contradictions. We know well that the imaginary uncertainty exists in mind of the designers. The imaginary complexity occurs due to unknown of the design actors and ignorance of the interactions between FRs and DPs.

Let us consider a design process of a re-design product with having 4 FRs that we must satisfy as shown as following:

$$FR_1 = X_1 DP_1 + X_3 DP_3 \tag{6.1}$$

$$FR_2 = X_1 DP_1 + X_2 DP_2 + X_3 DP_3 \tag{6.2}$$

$$FR_3 = X_1 DP_1 \tag{6.3}$$

$$FR_4 = X_1 DP_1 + X_2 DP_2 + X_3 DP_3 + X_4 DP_4 \tag{6.4}$$

At a glance, if we consider only one equation or few equations such as Equation (6.1), (6.2), or (6.4) by example, we will find no mathematical solution to resolve the design elements. Then we would consider that it is a complex design. On the other hand, if we gather those equations together and write them as a design matrix, it will be as shown in Equation (6.5).

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & X_3 & 0 \\ X_1 & X_2 & X_3 & 0 \\ X_1 & 0 & 0 & 0 \\ X_1 & X_2 & X_3 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix} \tag{6.5}$$

At this time, one might recognize that it is a coupled design due to the incorrect sequence. [Lee 2003] states that the complexity can be eliminated by identifying the structure of design matrices and follow the correct sequence dictated by design matrices. From Equation (6.5), we have to rewrite the functional relationship between FRs and DPs. In this case, we re-order first the sequence between FR_1 and FR_3 of the design matrix, as shown in Equation (6.6).

$$\begin{Bmatrix} FR_3 \\ FR_1 \\ FR_2 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & 0 & 0 \\ X_1 & 0 & X_3 & 0 \\ X_1 & X_2 & X_3 & 0 \\ X_1 & X_2 & X_3 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix} \qquad (6.6)$$

Then we change the sequence between DP_2 and DP_3, as shown in Equation (6.9). As a result, this design is considered as a decoupled design. Therefore, it is no more a complex design.

$$\begin{Bmatrix} FR_3 \\ FR_1 \\ FR_2 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & 0 & 0 \\ X_1 & X_3 & 0 & 0 \\ X_1 & X_3 & X_2 & 0 \\ X_1 & X_3 & X_2 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_3 \\ DP_2 \\ DP_4 \end{Bmatrix} \qquad (6.7)$$

With this method, we see well that each operation resolves at least one FR and it consequently resolves the following sequence of FRs. This recursive process allows us to reduce the complexity of design problem until the design is complete. Although the problem is an imaginary complexity, instead of using trial-and-error process to find the right sequence, the integrated design gives us the problem-solving without knowing the design matrices at the beginning.

However, in the real design project, there are not only few FRs but numerous FRs that we have to satisfy. One FR can also be decomposed into many levels as a hierarchical design to precise details of the design. Thus, we postulate that if the design actors work in the notion “just need”, then there ought to be at least one actor who is capable to resolve at least one FR by himself (if the design actors can resolve the FRs by themselves, it signifies that the design problem is in fact an uncoupled design). Then, that actor will give such information, which s/he can evaluate, into the share database. At this time, other actors will take such new information into account and resolve the problem step by step.

In order to describe the method of solving the problem of imaginary complexity, let’s consider Equation (6.5). Suppose that the assembler concerns the variable X_1, the mechanician concerns the variable X_2 and X_3, while the manufacturer concerns the variable X_4 as presented in Figure 6.1. In order to solve the imaginary complexity, the actor, who concerns only one FR, must propose a solution to the design team. In this case, the assembler must propose first a solution to satisfy the FR_3. This contribution of the assembler will enhance the mechanician to find the solutions that satisfy the FR_1 and the FR_2. As a result, the manufacturer will be able to propose a solution that satisfies the FR_4. This approach permits the design team to solve step by step the problem of imaginary complexity although the design equation was not in order as illustrated in Figure 6.2.

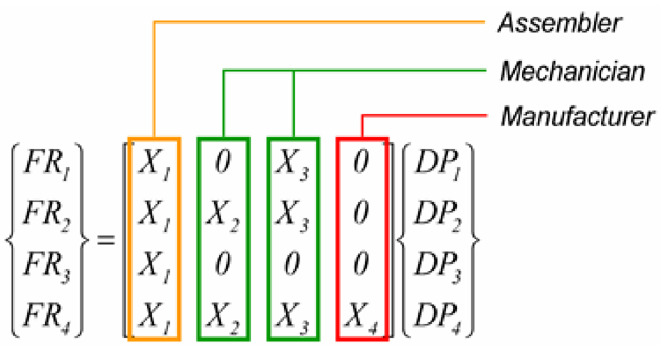


Figure 6.1 Example of a design equation

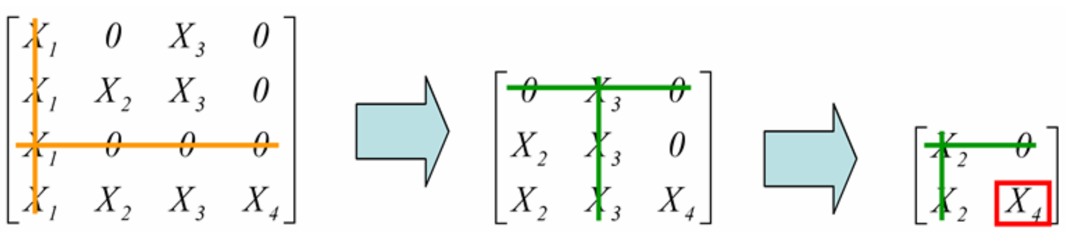


Figure 6.2 Method for solving the problem of imaginary complexity

This recursive process also continues in the low level hierarchical design until the design is complete. As a result, we can reduce the number of unsatisfied requirements, so does the complexity in the design process.

Let us consider a design process of a new product model, which is an innovative product, base on the complexity in Axiomatic Design. At the beginning of the design process, the design actors have only a few of information including the functional requirements (FRs) but they have no any solution (DPs) yet. Thus, the design matrix is unknown as written as following:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ \vdots \\ FR_n \end{Bmatrix} = \begin{bmatrix} ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? \end{bmatrix} \begin{Bmatrix} ? \\ ? \\ ? \\ ? \\ ? \end{Bmatrix} \tag{6.8}$$

With the support of the integrated design system, the actors can propose design solutions to construct a design matrix. According to the presented hypotheses and the notion of "just need", there must be at least one actor who is capable to resolve at least one FR. That actor has to give a solution to the design process that consequently allows the concerned actors to construct step by step a triangular design matrix, as presented in Equation 6.9.

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ \vdots \\ FR_n \end{Bmatrix} = \begin{bmatrix} X_{11} & 0 & 0 & 0 & 0 \\ X_{21} & X_{22} & 0 & 0 & 0 \\ X_{31} & X_{32} & X_{33} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{n1} & X_{n2} & X_{n3} & X_{n4} & X_{nn} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ \vdots \\ DP_n \end{Bmatrix} \tag{6.9}$$

Nevertheless, the design matrices are not always uncoupled or decouple but sometimes they could be a weakly coupled design. For example, let us consider Equation (6.10) as following:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \\ FR_5 \\ FR_6 \end{Bmatrix} = \begin{bmatrix} X_{11} & 0 & 0 & 0 & 0 & 0 \\ X_{21} & X_{22} & 0 & 0 & 0 & 0 \\ X_{31} & X_{32} & X_{33} & 0 & 0 & 0 \\ X_{41} & X_{42} & X_{43} & X_{44} & X_{45} & 0 \\ X_{51} & X_{52} & X_{53} & X_{54} & X_{55} & 0 \\ X_{61} & X_{62} & X_{63} & X_{64} & X_{65} & X_{66} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \\ DP_5 \\ DP_6 \end{Bmatrix} \tag{6.10}$$

This design equation seems to be a triangular design matrix. Actually there is a coupled design problem exists in this design equation, between the FR_4 and the FR_5. In this case, we consider that it is a weakly coupled design. The actors, who concern on this coupled design, cannot satisfy the corresponding FRs by themselves. However, the integrated design system permits the design actors to dialogue, to discuss on the design problem for finding an acceptable solution. In this case, the actors who concern the variable X_4 and X_5 have to discuss in order to solve the coupled design problem of weakly coupled design as represented in Figure 6.3.

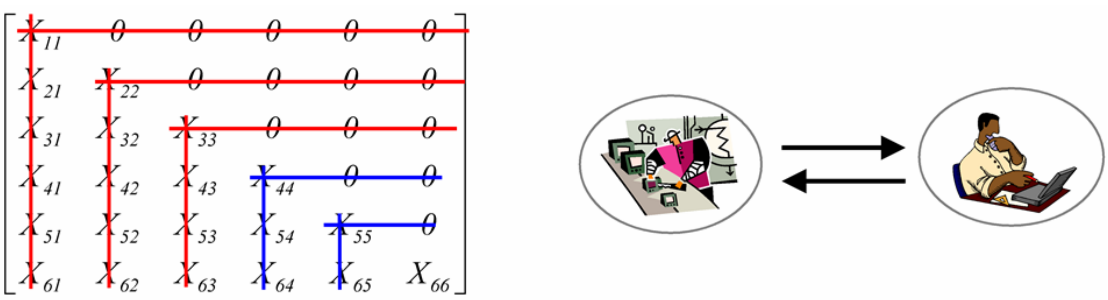


Figure 6.3 Interaction between design actors for solving a design problem

Our proposition of the integrated design permits the design team to resolve the problems of uncoupled, decoupled, and weak-coupled design. If the design matrix is a fully coupled design, changing any DP will affect the others and it can not be solved by the integrated design. In this case, the best way is to develop a new system to replace the coupled design with an uncoupled or decoupled design. One of the most well-known approaches is TRIZ (Theory of Solving Inventive Problems). The main objective of TRIZ is to evolve the system toward ideality by overcoming contradictions. There are numerous problem-solving patterns and tools of TRIZ that are developed and revealed in [http://www.triz-journal.com].

In any design project, the decisions at the highest level of FRs and DPs hierarchy have a profound impact and the viability of the project, including cost and time. If a coupled design is introduced at the highest level, it cannot be overcome by lower level design decisions. The system is also coupled and difficult to improve. We present here an example of designing a computer desk, represented in Figure 5.2. At the highest level, we begin with a set of functional requirement (FRs) as following:

FR_1 = Support a monitor

FR_2 = Integrate a place for a computer case

FR_3 = Support a keyboard and a mouse

FR_4 = Have a shelf for placing gadgets

This list of FRs characterizes the need of functions of the product, which are extracted from the customer's needs (CAs). In addition, design constraints (Cs) can be defined to the FRs. Constraints affect the design process by defining a bound of the acceptable design or solutions, and being references in the design evaluation. We can define a set of constraints as below:

C_1 = the maximum size of the monitor will vary from M_{min} to M_{max}

C_2 = the maximum size of the case will vary from CC_{min} to CC_{max}

C_3 = the maximum size of the keyboard will vary from KB_{min} to KB_{max}

We can now conceptualize a set of design parameters (DPs) to satisfy the FRs. In this example, the design team may propose DPs as following:

DP_1 = A top plate

DP_2 = A vertical plate and a shelf plate

DP_3 = A horizontal wide plate

DP_4 = A shelf plate

These DPs outlines the abstract of the product. We can now develop a design matrix for the high level design by given information as shown in Equation 6.11.

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & 0 & 0 \\ X_1 & X_2 & X_3 & 0 \\ X_1 & 0 & X_3 & 0 \\ 0 & X_2 & 0 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix} \tag{6.11}$$

From this equation, we have found that the design is a decoupled design. It means that the design achieves independent controls of all four FRs. Since this equation is a decoupled design, we should make clear that we do not create time-independent imaginary complexity by changing the sequence to be in order. To illustrate the decoupled design, we can re-order the sequence of the design matrix as following:

$$\begin{Bmatrix} FR_1 \\ FR_3 \\ FR_2 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & 0 & 0 \\ X_1 & X_3 & 0 & 0 \\ X_1 & X_3 & X_2 & 0 \\ 0 & 0 & X_2 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_3 \\ DP_2 \\ DP_4 \end{Bmatrix} \tag{6.12}$$

Note that in the design process, the proposed DPs that satisfy the FRs sometimes have constraints, which limit the bound of admissible solutions, as shown in the previous example. Some solutions are admissible while some solutions are not. Therefore, each actor must verify the proposed solution whether it effects those constraints or not. This condition brings into the notion of “just need” that “each actor has to contribute the constraints that s/he can prove..., the actor must be able to prove that, what he says is necessary to take into account of such constraints”.

Once the highest level of FRs and DPs has been developed, the design may need to further decompose the FRs and DPs into lower levels to define more specific requirements. This decomposition can be performed by mapping between functional domain and physical domain, which is called *zigzagging* (see 3.2). The process of decomposition has to be continued to lower levels of hierarchical design until the design is complete.

In order to facilitate the design actors to realize the constraints and to understand the information established by the previous design actors, the system must notify the concerned design actor(s) such relevant information and constraints. Thus, CoDeMo provides the design team the method of data translation and data propagation. The data translation interprets the information of one actor which is relevant to another actor(s) into a comprehensible form. The data propagation propagates that relevant information to the corresponding actors. These two methods have been presented in (4.3.8).

6.3 Interaction between actors

Figure 6.4 presents the well-known iceberg model of one's knowledge. There are four types of knowledge representation i.e. we know what we know, we know what we do not know, we do not know we know, and we do not know what we do not know. The cognitive sociology affirms that individuals have knowledge in their mind but it is very difficult that one can know everything what s/he knows, as shown as the right of the iceberg. It is defined that the implicit and the explicit knowledge of one person as represented by the top of the iceberg in the segment of *conscious/knowledge*. On the other hand, there is a vast knowledge that one does not know and even does not know what s/he does not know as presented on the left of the iceberg. Therefore it is not surprising that the imaginary complexity might occur any time during the design process.

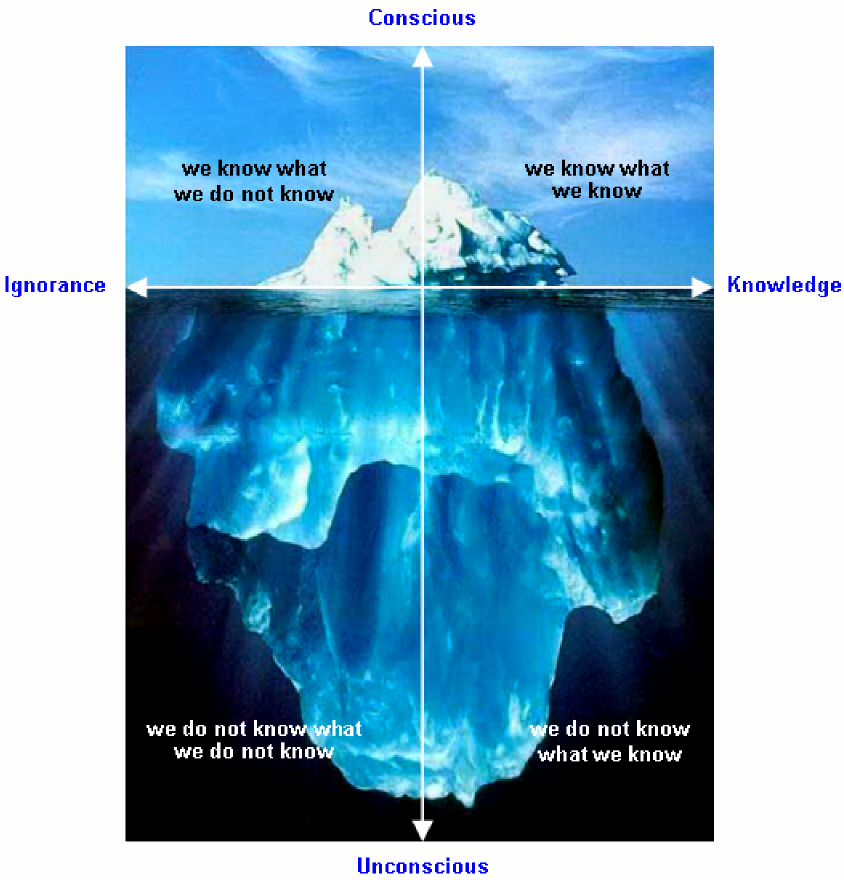


Figure 6.4 Iceberg model representing one's knowledge

The imaginary complexity rises because the designers lack of information. The simple solution to reduce this complexity is that to make the designers know what they ought to know. Let us concentrate on the segment of *conscious/knowledge*, we know what we know, this segment can be compared as '*product model*' (see 4.2.1) that stores the product information during the design process. By the view of the design team, this segment stores knowledge that is in mind of individuals, also known as '*knowledge model*'. As long as an individual has no interaction with any other, s/he can not enhance the knowledge model. Therefore, the imaginary complexity still remains. On the other hand, as far as more and more s/he involves in the integrated design process, more and more knowledge is shared and taken into account to develop the products.

As introduced in (4.3.2) that the integrated design process is divided into two phases. We present continually the interaction between the actors during these two phases. It presents how the design actors deal with design problems and how to couple DPs to the corresponding FRs.

6.3.1 First design phase

[Belloy 1994] proposes the concept of functional surface. This concept permits the technologist to integrate his/her knowledge into the product by using the technological solutions, i.e. the choices of solutions, the constraints of kinetics, tolerance, dimension, and assembly. This aims to initiate minimal information to recognize the functional surfaces of the product [Tollenaere et al 1995]. The technologist permits the other actors to retrieve this initial information for evaluating the product in their view.

In comparison with [Belloy 1994], the first design phase of this study is facilitated by the conceptual design. At the beginning of this phase, the actor who concerns global form and aesthetic of a product, such as dimension, texture, color, etc., must propose a conceptual design of the product. The conceptual model is normally handled by a CAD system and should be manipulated with primary specifications and functional requirements. Consequently, that actor has to output the conceptual design into a universal standard format.

In this study, we apply a standard format file such STEP (Standard for the Exchange of Product Model Data) [http://www.steptools.com], which is compatible with various CAD systems. We use a CAD modeller to transform the conceptual design into a STEP format file which is a neutral file. At this time, we ask the technologist to set off the initial information by transforming the neutral file into the collaborative environment as illustrated in Figure 6.5. This initial information comprises the global form and the default dimensions of the product that are used as a starting point for the integration of knowledge in the second design phase.

The first phase would be accomplished when the functional surface of parts have been defined. The conceptual design provides the shape and the propositions that explicit the functions and the mode (style) of the product as illustrated by example of a computer desk in Figure 5.2. However, the conceptual design does not provide any solution about assembly, mechanical behavior, or manufacturing characteristics. For example, how much weight is the desk capable to resist; how to assembly or what kind of fasteners should be applied to fasten the parts, etc. In order to determine DPs and define values of their characteristics, we need the concerned design actors to participate the detail design of the product and to introduce their information and constraints into the design process.

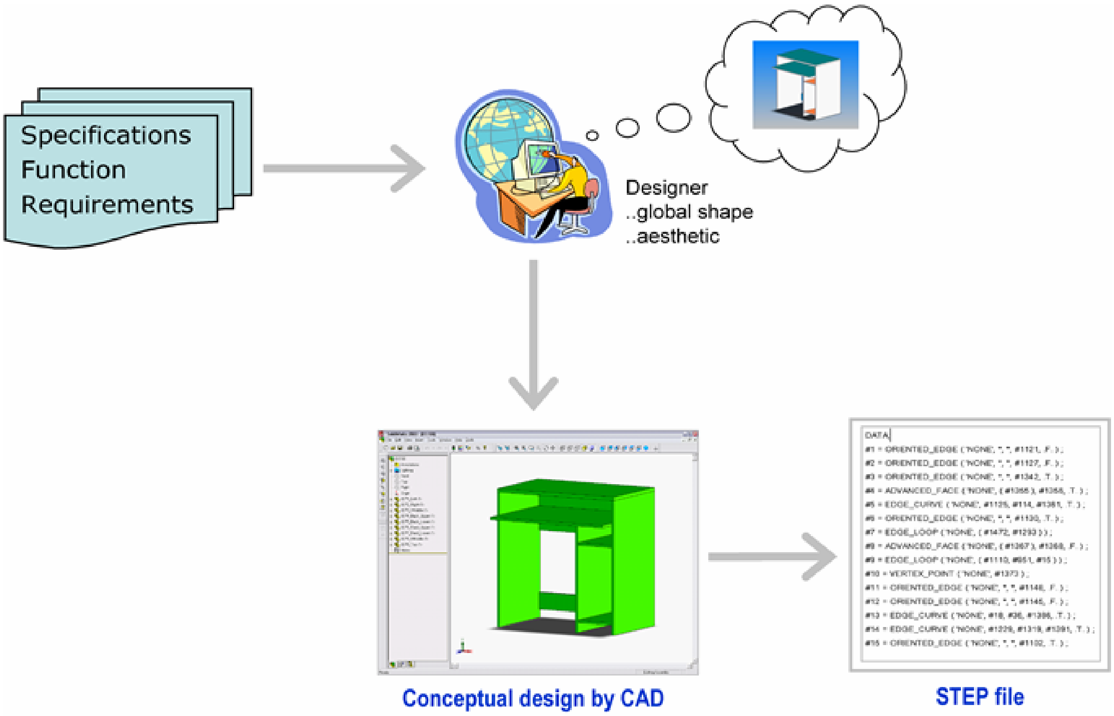


Figure 6.5 Work flow of the first design phase

6.3.2 Second design phase

During this phase, the design actors are asked to participate in the design process for coupling DPs to the corresponding FRs and also for defining values to the DPs. The design actors contribute their information, constraints, and points of view to the product step by step. In fact, this phase begins as soon as one has enough relevant information to realize his/her own evaluation of the product. This overlapping is flexible. The design actors can perform their tasks as parallel without waiting for the first design phase finished the functional recognition. We propose the collaborative environment of the design process as illustrated in Figure 6.6.

During the design process, the design actors have to maintain the style and the functions of the product and also to take into account the design in term of cost and quality. CoDeMo is now asked to run the collaborative system using the multi-view concept (see 4.3.6) that provides a trade view for each discipline. It provides methods and models to characterize the product with specific information, and gives all actors to recognize dynamically the product in common views. In this study, we take into account three principal trades that are assembly view, mechanical view, and manufacturing view.

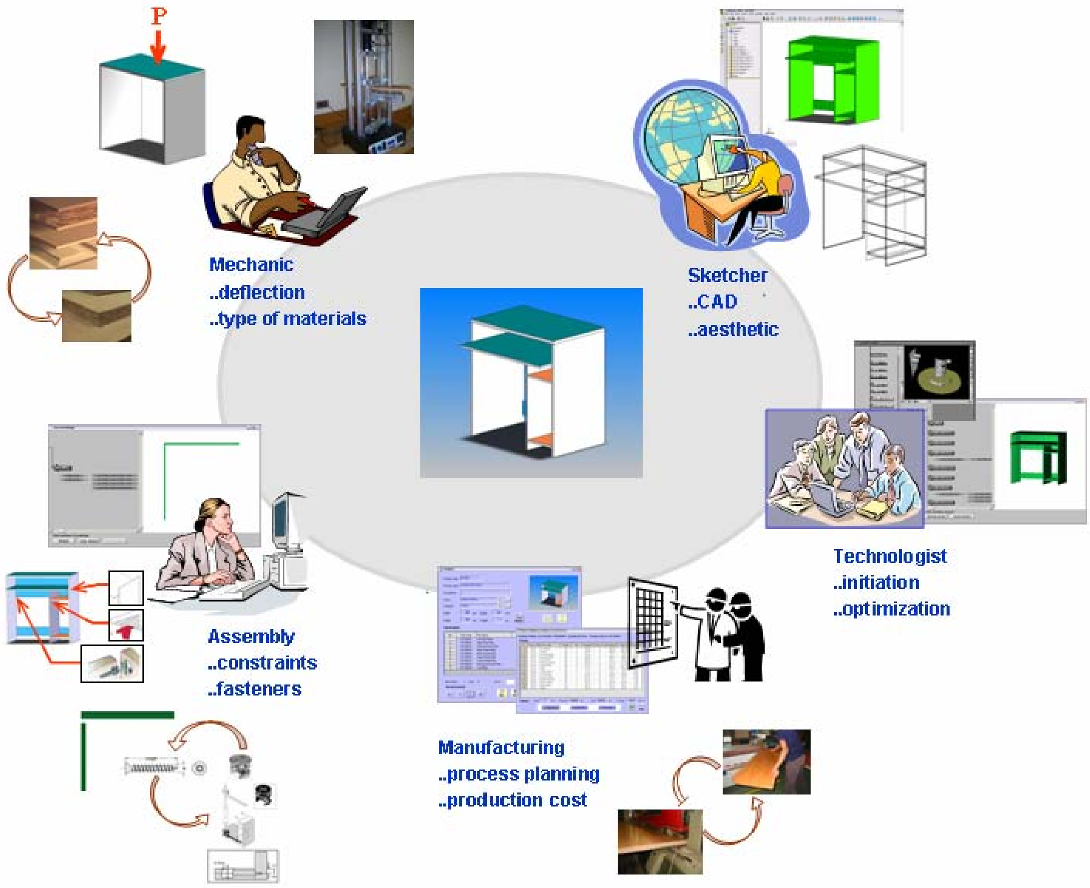


Figure 6.6 Collaborative environment of a design process

Assembly view

In this view, the assembler concerns to choose an appropriate solution to fasten each pair of parts. Of course, there is always more than one solution for fastening a pair of parts but the assembler must choose the most appropriate solution depending on the FRs and available fasteners that we have. The chosen solutions will be DPs of the corresponding FRs of the assembly view, which might affect the other FRs. Of course, the different fasteners used in the same product have to be homogeneous. Figure 6.7 shows by examples some alternatives of how to fasten the parts of the conceptual product.

In the case (1), a set of cam-steel dowel, is proposed to use to fasten the two vertical plates (the left part and the top-back part). This choice can satisfy the FR that requires high quality fasteners with good resistibility. Otherwise the assembler might propose fasteners (a), a screw or a wood dowel that is easy to assembly and less expensive than the previous one in order to economize the cost. Nevertheless, a screw has much less resistant than a set of cam-steel dowel and is not aesthetic when its head appears at the surface of the vertical plate. Yet, the assembler might propose a wood dowel instead of the previous two. It gives the aesthetic and the least expensive. It gives also more resistant than a screw but less than a set of cam-steel dowel. However, the drilling and assembly operation are not as simple as using a confirmat screw.

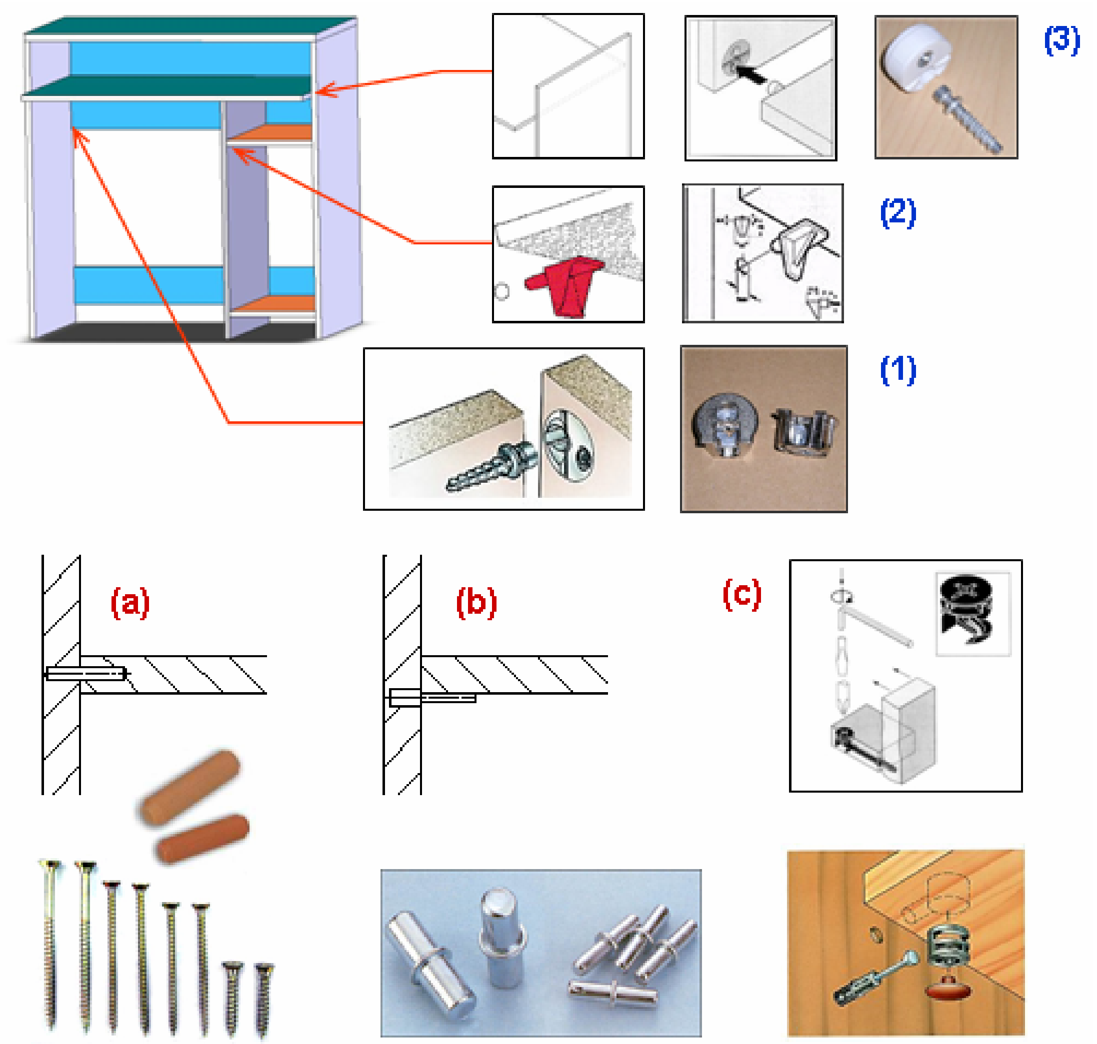


Figure 6.7 Examples of choosing assembly solution

In the case (2), if one of the FRs requires the horizontal plate (the small top part) being detachable, a plastic shelf support is proposed to attach to the vertical plate (the middle one) to support the horizontal plate. However, the fastener (b), a steel shelf support, which has better quality but more expensive, would replace a plastic shelf support if the FR also requires to support a heavy load. One can notice that these two supports can never be used to hold two vertical plates such as the case (1). In contrast of being detachable, the assembler might propose to use a wood dowel to fix those two parts together. It costs less expensive than those two supports. On the other hand, this solution induces another load to manufacturing process. The manufacturer has to drill both of two plates instead of only the vertical plate. In addition, using wood dowels needs the parts, which are concerned to the inserted dowels, to be assembled at the same time. In this case, if we fasten the two shelf-parts (the middle-horizontal and lower-horizontal parts at the right) to the right-side part and the middle-vertical part by using wood dowels, it would be difficult for customers to place the parts in the correct position. On the other hand, using supports is simple. The customers just put them in the holes and place the shelves.

In the case (3), the horizontal wide plate is attached to the vertical plate. A set of plastic cam-steel dowel is proposed to fasten these two plates together. However, if the FR requires durability, this fastener might be not suitable for this large assembly. A strong mechanical fastener is needed. The assembler might propose the fastener (c), a knock-down kit, to fasten for this permanent joint. Again, it costs more expensive than a set of plastic cam-steel dowel.

It is clearly that a different solution gives a different affect. One might be less expensive but does not satisfy the aesthetic. One might satisfy the FR of the aesthetic but affects to another FR that requires the ease of assembly or manufacturing. One might satisfy the FR of having a good resistibility or being durable but the cost is expensive. A chosen solution gives to the product a different cost, quality, and also manufacturing process. Therefore, to consider what solution should be applied, we do not concern only the customer's requirements but also we have to respect the agreements of the other trades.

Note that the assembler has to define later the details of the fasteners e.g. size, diameter, etc. when the other design actors have given the additional relevant information. For example, when the mechanician has chosen the thickness of the parts, the assembler has to choose consequently the diameter of dowels. Sometimes, the assembler may be asked to change the assembly solutions if the choices of the other design actors that are more significant affect his/her proposed solutions.

<u>Mechanical view</u>

Beyond the aesthetic design, the quality of the product is one of the crucial customer's requirements. Mechanics has to guarantee the durability and resistibility of the product. The objective of this view is to define the most appropriate material types and thickness of plates to the product. The mechanical view possesses information of materials as a part of his/her knowledge e.g. material types, physical and mechanical properties of materials – density, modulus of rupture, modulus of elasticity, etc. To run a mechanical test, the mechanical view needs to recognize the product structure and to retrieve default dimension of plates (if exist). However, the mechanical view must have enough relevant information, knowledge and constraints from the other actors that affect the evaluation e.g. available materials, required standard, the assembly solutions from the assembler, etc.

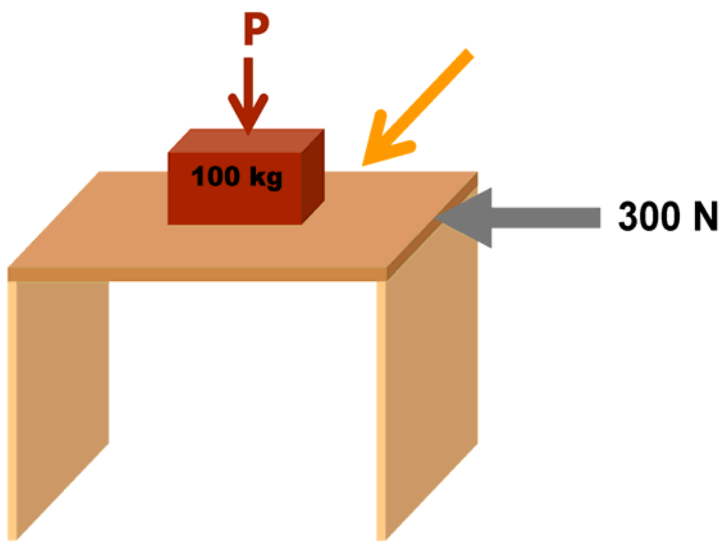


Figure 6.8 Example of mechanical testing

The deflection of the plates depends on a given load, respecting to the required standard. Figure 6.8 shows by example a mechanical testing of a table. According to the NF for Professional Furniture [NF Ameublement], which is the French furniture standard, it requires a test of strength and durability. A table is required to resist a load of 100 kg on the top, five accidental drops by tipping it over for testing the strength, and 10,000 lateral thrusts of 300 N for testing the durability. The constant values such 100 kg, 5 drops, 10,000 lateral thrusts, 300 N are constraints that must be concerned by the mechanician (an example of deflection test is demonstrated in Chapter 7).

After the test, the mechanician should be able to determine the DPs and its values that satisfy the FRs. S/he can accept or refuse the default thickness (if exist) and the given materials. Otherwise, s/he can also define new material types and thickness for the plates depending on the given FRs. Nevertheless, the assembly structure also affects the mechanic's decision. For example, the assembler is considering to define an assembly structure for a shelf as shown in Figure 6.9. In case of using the solution (a), the vertical plates will resist the load more than the top plate. Mechanics must concern the material type and thickness of the vertical plate. On the other hand, if the mechanician chooses the solution (b), the load will act on the top plate. The plate will transmit the load to the edge-surfaces, which are fastened to the side-parts, and the fasteners. In this case, the mechanician must choose a material with high property of modulus of rupture and elasticity or having more thickness for the top plate. Otherwise, it is risk that the edges of the top plate might be shredded due to a heavy load. The assembler may possibly need to use high quality fasteners to fasten the top plate. If the mechanician does not agree with the assembly structure by a failure of the test, s/he can then ask the assembler or one who concerns this task to review the definition of product structure.

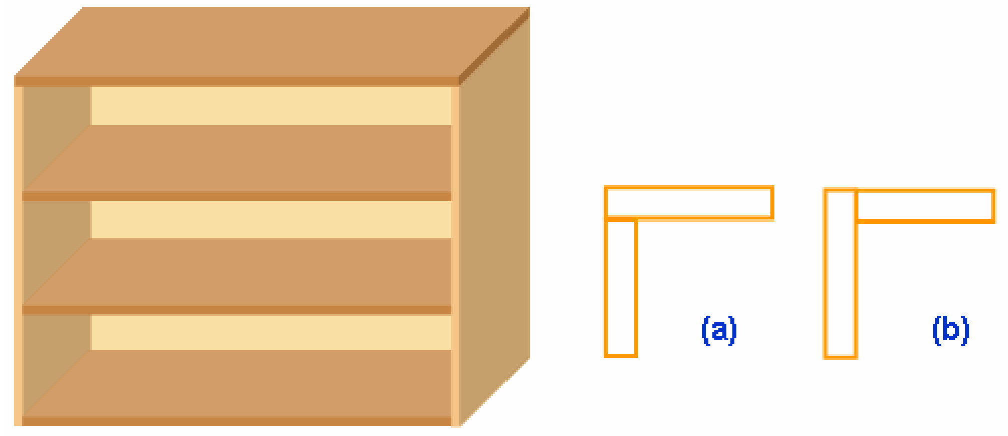


Figure 6.9 Example of defining an assembly structure

<u>Manufacturing view</u>

The manufacturer plays an important role in the design process. The primary objective is to estimate the manufacturing cost, material cost, and to evaluate the producibility of the product following the design. To evaluate the design, the manufacturer has to recognize first the manufacturing constraints which can refer to the ability of manufacturing e.g. the capacity of machines, human resource, available/required tools and machines, etc. and also manufacturing criteria such as the ease of manufacturing, the manufacturing cost, etc. Consequently, the manufacturer also has to take account the previous DPs and constraints of the other actors, which affect the FRs in the manufacturing view.

Choosing a different assembly solution gives a different manufacturing process and also a different manufacturing cost. For example, the case (2) in Figure 6.5, if the assembler chooses a plastic or steel shelf support for the shelf, the manufacturer just drills only one hole for one support. On the other hand, if the assembler chose a wood dowel to fix the shelf, the manufacturer must drill two holes for one support, one at the vertical plate and the other one at the shelf (horizontal plate). Though a wood dowel is less expensive, but it raises the manufacturing cost and time. Therefore, the manufacturer can accept the design if it satisfies his/her FRs and constraints. Otherwise s/he can negotiate with the assembler, and/or those who concern this task, to revoke the assembly solution or to adjust their DPs and its values. This negotiation is to optimize the design. In this study, we develop an application named DAPP, *Database Application for Production Planning*, to be a specific tool in the manufacturing view. CoDeMo allows the manufacturer to use this specific tool to estimate the manufacturing cost and to manage the process planning. To perform this evaluation, the manufacturer needs the relevant information from the design team and the system.

The internal actor occupies the tasks of translation and propagation by translate the relevant information contributed from other trades and propagate to the manufacturer. For example: details of the assembly solutions from the assembler, the material types and thickness of the plates from the mechanic, tolerance and characteristics of the plates from the frame view, geometrical information from the geometry view, etc. As soon as the propagation process is complete, CoDeMo will output this integrated information to DAPP. The manufacturer will manipulate the received information and his/her knowledge into DAPP. With the contribution of the manufacturer, DAPP can output the cost estimation of the product and also plans the manufacturing process for each part. Yet, DAPP can not only estimate the manufacturing cost and plan the manufacturing process but it is such a database that used for collecting the information for production. Figure 6.10 shows by example the functions of DAPP. However, this specific tool is presented in Chapter 7.

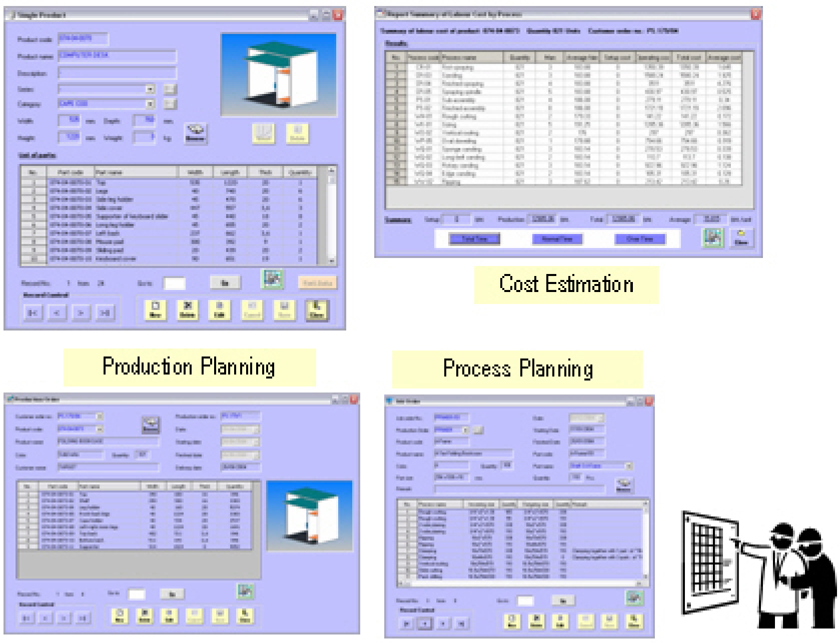


Figure 6.10 Example features of DAPP

We can summarize the interaction and evaluation process of any actor by presenting as a model in Figure 6.11. To evaluate the design of an actor, s/he first retrieves the initial information from the shared database contributed by the design team. As soon as s/he has enough information to define DPs and its attributes, the evaluation is then processed. If the result satisfies the FRs, the actor will send it to the shared database as new data of the product and the system will transmit the result to the other actors who may concern. On the other hand, if the result does not satisfy the FRs, the actor must adjust DPs values that satisfy the FRs. Otherwise, if the actor proves that the known DPs constrain his/her DPs or obstruct the evaluation, s/he has to ask the concerned actor(s) to negotiate on the problem by revoking or adjusting that constraint(s).

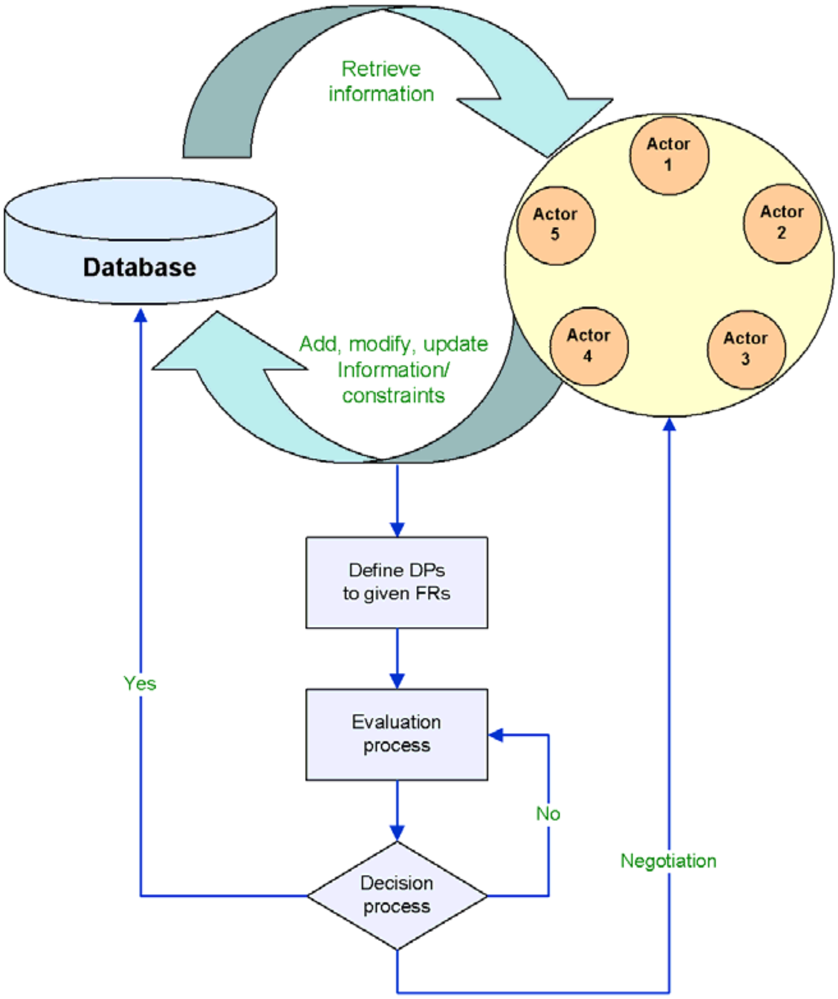


Figure 6.11 Model of interaction and evaluation process of an actor

From the model in Figure 6.11, one can notice that although the design actors contribute their constraints as soon as possible to the design team, this model still creates some design iterations. The source of the design iterations occurs due to the single evaluation of each design actor. If the given solution fails to satisfy the FRs, the design concerned actor(s) will be asked to re-discuss on the problem and to negotiate until they find the acceptable one. What we project to develop further is to give the design actors a concept of '*multi-variant*'. It means that the design actors can propose alternatives to the team more than one solution. This perspective could reduce a number of re-design processes and enhances the design team to evaluate and to choose the most appropriate solution for the design process.

6.4 Summary

This chapter has presented a method to reduce the time-independent imaginary complexity which is based on the Axiomatic Design. The method is that to make all design actors in the design team recognize the relevant information, knowledge, and constraints of each other as much and as soon as possible during the design process. The more they pose their problems, the more problems will be solved and the more they give constraints and information, the more imaginary complexity will be eliminated. With this method, they can characterize the product and evaluate the design step by step. As a result, they can identify the design matrices by establishing corresponding DPs and its values to the proposed FRs.

To establish the DPs and its values, the system brings the design actors to be presented in a virtual meeting room. This permits them to share and to exchange knowledge and constraints to characterize the product. As soon as one has enough information, s/he will evaluate the design in his/her own view and then contribute the result back to the design team and the shared database. The evaluation process keeps continuing until achieve the design with the given criteria. The demonstration of the interaction between the design actors and the evaluation process will be presented in Chapter 7.

Part III

Integrated Design System

Chapter 7

Integrated Design System

This chapter presents the integrated design system that has been developed by the *integrated design* team of the laboratory G-SCOP. The Cooperative Design Modeller, *CoDeMo*, is based on a client-server system. It creates a virtual meeting room that permits the design actors in the design team to perform the design activities together with distant and synchronous or asynchronous access. This chapter demonstrates how the design team applies *CoDeMo* to perform the design tasks and how to apply their specific application to evaluate the design.

7.1 Introduction

Most of current CAD systems are based on a geometrical modeller and have a main role in the current design practice. Geometry is considered as a type of universal feature, which can be easily shared for the team. For that reason, they develop CAD systems as geometric modeller to support the demand of market. Such geometric modellers are developed to principally facilitate the designers to draw a product and parts, and to picture the conceptual design. The utilization of a CAD system for drawing a model does not faster than traditional methods, but the modification is much easier. However, they have not taken account of the integrated aspects. They just did not realize that there is no person who is in charge of the geometry, but it is just a consequence of the design. As a result, they require afterward contributions from the concerned persons to define completely the geometrical product model. Many researches realize this problem and have tried to solve by taking into account trade knowledge and constraints into the design process [Noël et al 2003]. In this study, we develop an integrated design system based on the concurrent engineering approach. We present in this chapter the concept of CAID (Computer Aided Integrated Design) system that allows the design actors from all trades to work together. The integrated design tool "Cooperative Design Modeller" or *CoDeMo* is a result of the studies of Elsie Chapa Kasusky [Chapa Kasusky 1997] and Lionel Roucoules [Roucoules 1999] who have implemented the concept of product model, and have developed the dynamic representation of the data model and the dynamic creation of the knowledge model.

The main objective of CoDeMo is to accomplish the inadequate in CAD systems. [Prasad 1996] states that a set of network traits that enable integration must consist of: open system traits, shared traits, client/server traits, and gateway and protocol. CoDeMo is based on a client-server system. It gives an access to the design actors to work on the same project, which is stored in the shared database, as a multi-actor system. CoDeMo permits the design actors (clients) to connect to the shared database via a formal network. The formal network is based on RPC (Remote Procedure Call) and CORBA (Common Object Request Broker Architecture) architecture, the architecture of distributed object – presented in [Radulescu 2005], between the client process and the server process. We used the ILOG libraries [ILOG] and C++ programming language to develop the design environment. CoDeMo actually takes place in a collaborative environment as shown in Figure 7.1.

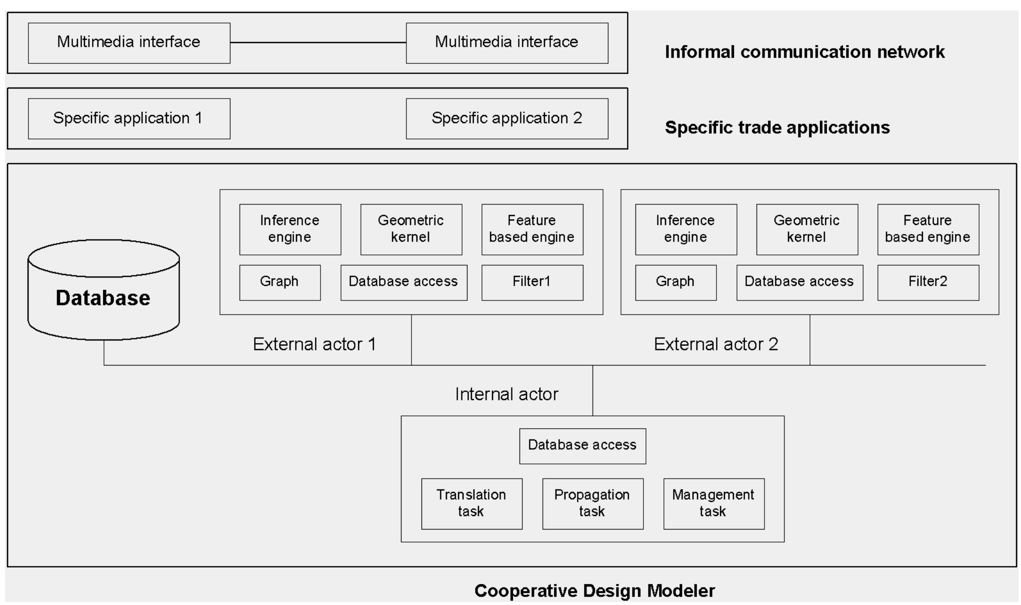


Figure 7.1 Structure of CAID modeller [Roucoules and Tichkiewitch 2000]

This collaborative environment consists of three parts: the CoDeMo system itself, the specific trade applications, and an informal communication network. *CoDeMo* is the tool that creates a virtual meeting room by providing a formal communication network, a shared database, and methods that enhance the design actors to participate in the design process together. A *specific trade application* is a part of the design environment. It is developed to facilitate a specific trade actor to realize a specific task(s). A specific trade application needs an initial data, which is normally obtained from CoDeMo, to evaluate the design in a specific task. After the evaluation, the specific application provides the result to CoDeMo. To exchange such information, it is necessary to study the standard exchange format between each other. The *informal communication network* is needed when a normal dialog is required to arrange a design conflict(s). The normal dialog means that the informal network has to permit the design actor to write, to talk, to sketch, and/or to watch each other so that they can discuss on the problematic. There are a lot of commercial products such as Instant Messengers, Net Meeting, AREL, etc., who permits this dialog.

The architecture of CoDeMo and the structure of a shared database have been well described in [Tichkiewitch 1996], in methodology and system of holonique design [Chapa Kasusky 1997], and [Roucoules and Tichkiewitch 2000].

7.2 Initiate a design project

We present in this chapter an example of the design of furniture made of particleboard and medium-density fiberboard. Step by step, it allows the reader to understand exactly what CoDeMo can do in this field and what are the internal actions and expectations from the different actors. To run a new design project, the initiator has to establish the formal exchange network by launching the ILOG Port Mapper (ilbpmap) process and ILOG Broker Logical Mapper (ilblmap) to enable the RPC communication, as represented by icons in Figure 7.2. As well as the initiator, the actors who connected to the network also have to launch these processes. We consider here that the manager of the network has launched the ILOG Broker server process. This server process actually manages the shared database of every project in current design. It also manages the product (data model) structure and executes internal tasks, the server process is known as the *internal actor*. This process is only concerned by the manager, and is running in a background task. Afterward, every design actor, who wants to participate to the design process, must launch the ILOG Broker client process to connect to the design project. This process provides GUI to the clients and manages the remote calls from the clients to the server process. The design actors can connect to the design project as soon as the initiator has launched the server process.

Figure 7.2 Shortcut commands for launching process of server and client

A first interface is proposed as given in Figure 7.3 (a). In the case of new design project, the initiator must create a new one (by click on the 

button) and then provide a name and a password to the project. The password is relative to one design actor and one project. It means that the same design actor can have different password for different project, and different actor can have different password for the same project. In order to connect to the network, a design actor has to be authorized by the manager, and is recognized as being able to represent some specific trades. This is to allow the design actor to launch some specific views. For the existing projects, as soon as the client process has been launched, the design actors can choose the desired project, input the password as represented in Figure 7.3 (b), and then choose a corresponding view which they want to connect as represented in Figure 7.3 (c).

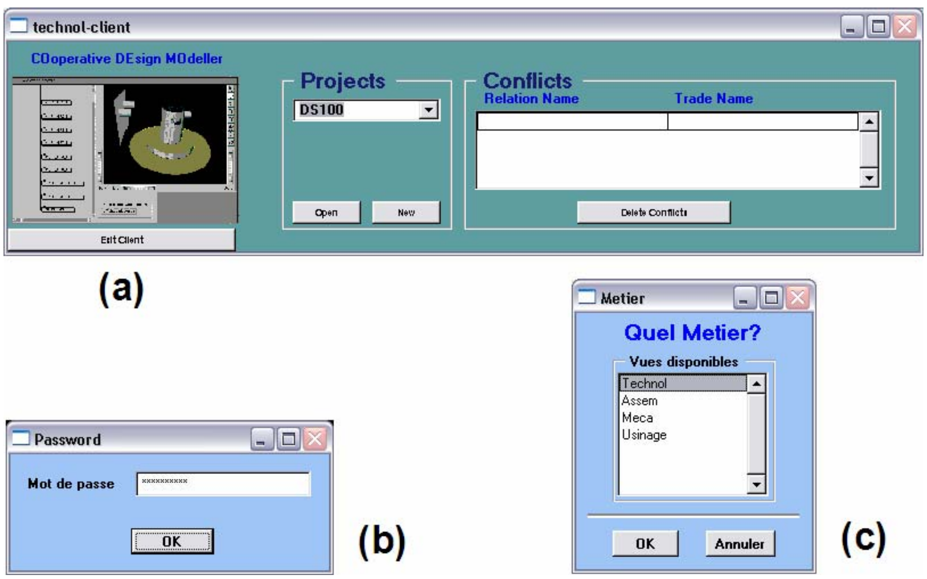


Figure 7.3 Initiation of a new project or access to an existing project

Once the design actors have chosen a project, CoDeMo brings them into the collaborative environment and gives them an access to connect to the shared database. CoDeMo creates specific trade views and common views for design actors, as represented in Figure 7.4: a trade view of technologist, frame view and geometric view. Note that one trade view may be used by several actors who are in the same domain of competence. The initial status of a new project in specific trade views and common views has empty information except the technological view that has an initial component. To continue the design process of this project, the initial design and the detail design phases are presented in the following sections.

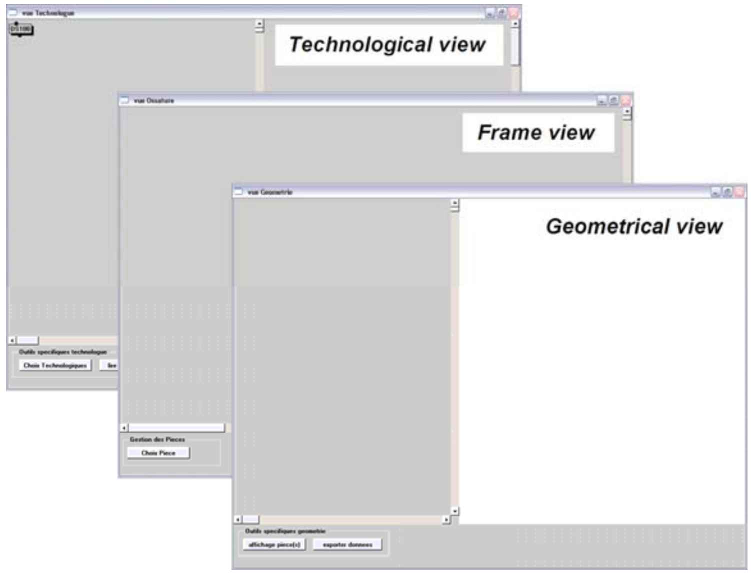


Figure 7.4 Initial status of a trade view and common views

7.3 Demonstration of initial design phase

Let us consider a new computer desk named *DS100*, as represented in Figure 5.2. In the beginning of the design process, the actor who concerns the specifications of the product must propose the shape and the meaning of aesthetic of the desk, which takes into account the global dimensions, color, surface attributes, in a 3D geometric CAD system. This virtual model is an empty point for the CoDeMo system. To facilitate CoDeMo to recognize the geometric data of the conceptual design, in this study, we ask a CAD system to transform a CAD file of the conceptual design into a standard format file, *STEP-AP203*. This neutral file permits CoDeMo to capture geometric data and essential information of the conceptual design into the design process. Table 7.1 shows a part of the STEP-AP203 file of the conceptual design.

The technologist is considered to be the first actor to connect to the formal network as s/he is the actor who is in charge of the definition of functional surfaces, which determine the possibility to continue in the second phase. S/he has to initiate his task by translating the input STEP file into the shared database. From the specific trade view, the technologist translates the STEP file via his specific tool panel, as represented in Figure 7.5. In fact, this translation is done by the internal actor. We have developed the data translation method to recognize the schemas of the STEP file. The internal actor employs the data translation method to translate the STEP into the shared database and represents it in the form of product model (components, links, relations, and entities). It also employs the data propagation method (see 4.3.8) to transmit such information to the corresponding common views and trade views.

Table 7.1 Example of an AP203 STEP file

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (( 'STEP AP203' ), '1' );
FILE_NAME ('DS100.STEP', '2006-05-10T09:50:33',
   ( 'Kusol' ), ( 'PIMAPUNSRI' ), 'SwSTEP 2.0',
   'SolidWorks 2002296', '' );
FILE_SCHEMA (( 'CONFIG_CONTROL_DESIGN' ));
ENDSEC;

DATA;
#1 = ADVANCED_FACE ( 'NONE', ( #542 ), #543, .F. ) ;
#2 = EDGE_CURVE ( 'NONE', #224, #149, #550, .T. ) ;
#3 = VERTEX_POINT ( 'NONE', #554 ) ;
#4 = VERTEX_POINT ( 'NONE', #555 ) ;
#5 = EDGE_CURVE ( 'NONE', #1601, #495, #556, .T. ) ;
#6 = ORIENTED_EDGE ( 'NONE', *, *, #2456, .T. ) ;
#7 = ORIENTED_EDGE ( 'NONE', *, *, #501, .F. ) ;
#8 = EDGE_CURVE ( 'NONE', #392, #2457, #602, .T. ) ;
#9 = ORIENTED_EDGE ( 'NONE', *, *, #176, .T. ) ;
#10 = ORIENTED_EDGE ( 'NONE', *, *, #389, .F. ) ;
#11 = ORIENTED_EDGE ( 'NONE', *, *, #334, .T. ) ;
#12 = ADVANCED_FACE ( 'NONE', ( #607 ), #608, .F. ) ;
#13 = VERTEX_POINT ( 'NONE', #620 ) ;
#14 = ADVANCED_FACE ( 'NONE', ( #621 ), #622, .F. ) ;
#15 = EDGE_CURVE ( 'NONE', #3, #64, #628, .T. ) ;
#16 = VERTEX_POINT ( 'NONE', #633 ) ;
#17 = ORIENTED_EDGE ( 'NONE', *, *, #433, .F. ) ;
#18 = ADVANCED_FACE ( 'NONE', ( #654 ), #655, .T. ) ;
#19 = ORIENTED_EDGE ( 'NONE', *, *, #394, .T. ) ;
#20 = ORIENTED_EDGE ( 'NONE', *, *, #400, .F. ) ;
#21 = ORIENTED_EDGE ( 'NONE', *, *, #1946, .T. ) ;
#22 = EDGE_CURVE ( 'NONE', #502, #1691, #673, .T. ) ;
#23 = ORIENTED_EDGE ( 'NONE', *, *, #182, .T. ) ;
#24 = VERTEX_POINT ( 'NONE', #677 ) ;
#25 = ORIENTED_EDGE ( 'NONE', *, *, #2, .T. ) ;
#26 = ORIENTED_EDGE ( 'NONE', *, *, #122, .T. ) ;
#27 = EDGE_LOOP ( 'NONE', ( #1297, #848, #519, #491 ) ) ;
#28 = EDGE_LOOP ( 'NONE', ( #390, #1619, #124, #82 ) ) ;
#29 = EDGE_CURVE ( 'NONE', #388, #13, #688, .T. ) ;
#30 = CLOSED_SHELL ( 'NONE', ( #395, #486, #430, #2323, #196, #1 ) ) ;
#31 = EDGE_CURVE ( 'NONE', #16, #4, #705, .T. ) ;
#32 = ORIENTED_EDGE ( 'NONE', *, *, #8, .T. ) ;
#33 = EDGE_LOOP ( 'NONE', ( #2359, #1128, #529, #217 ) ) ;
#34 = EDGE_LOOP ( 'NONE', ( #6, #1979, #42, #351 ) ) ;
#35 = EDGE_CURVE ( 'NONE', #80, #371, #741, .T. ) ;
.
.
ENDSEC;
END-ISO-10303-21;
```

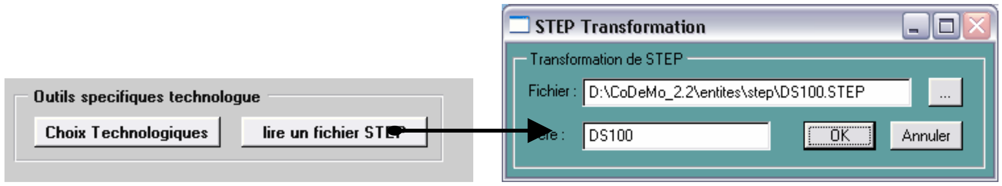


Figure 7.5 Specific tool panel of technological view

As a result, the product specifications and geometric data of the conceptual design have been transformed into CoDeMo database. Different from the study of [Belloy 1994], the technological view is mainly exploited to initiate the cooperative design process, as represented in Figure 7.6. Not only the technological view, this initial design phase also recognize functional surfaces of the product and presents in the frame view and the geometric view by a translation of the input STEP file. These common views are employed to describe the functional surfaces and characteristics of the parts. In the frame view, we may define the characteristics of materials used to cover the surfaces of the parts such as color, texture, type of material, etc. Figure 7.7 represents data model (product structure) of the product in the frame view after translated the STEP of the product.

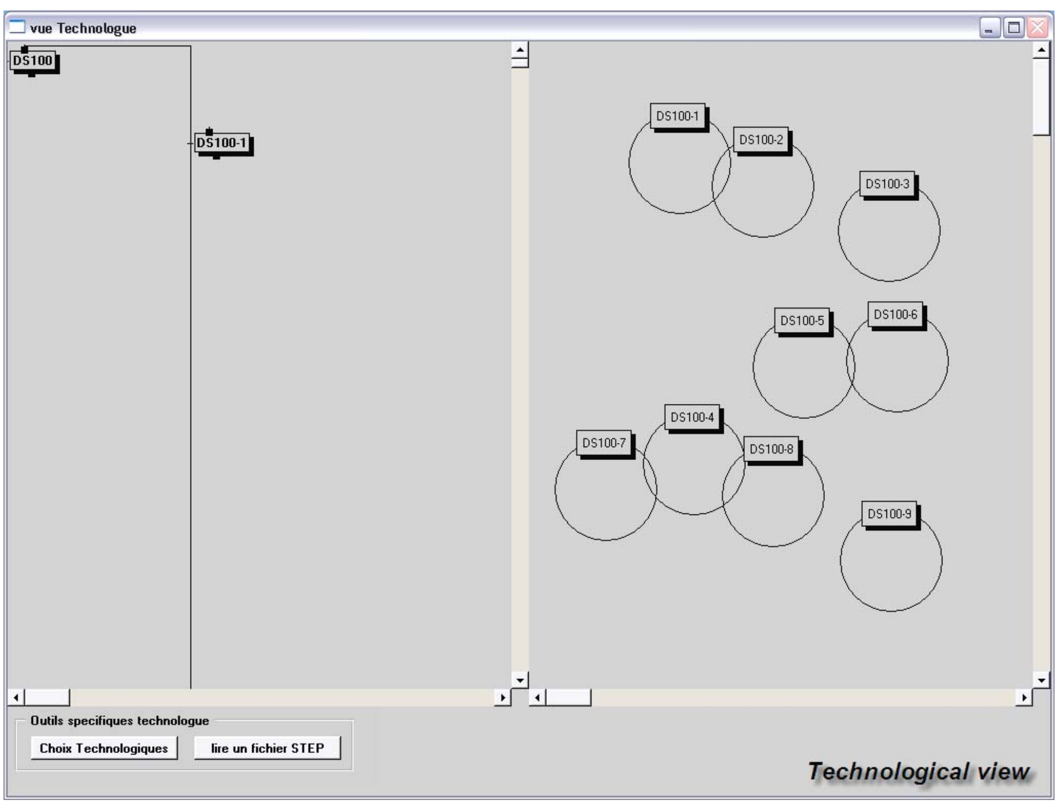


Figure 7.6 Initial information represented in technological view

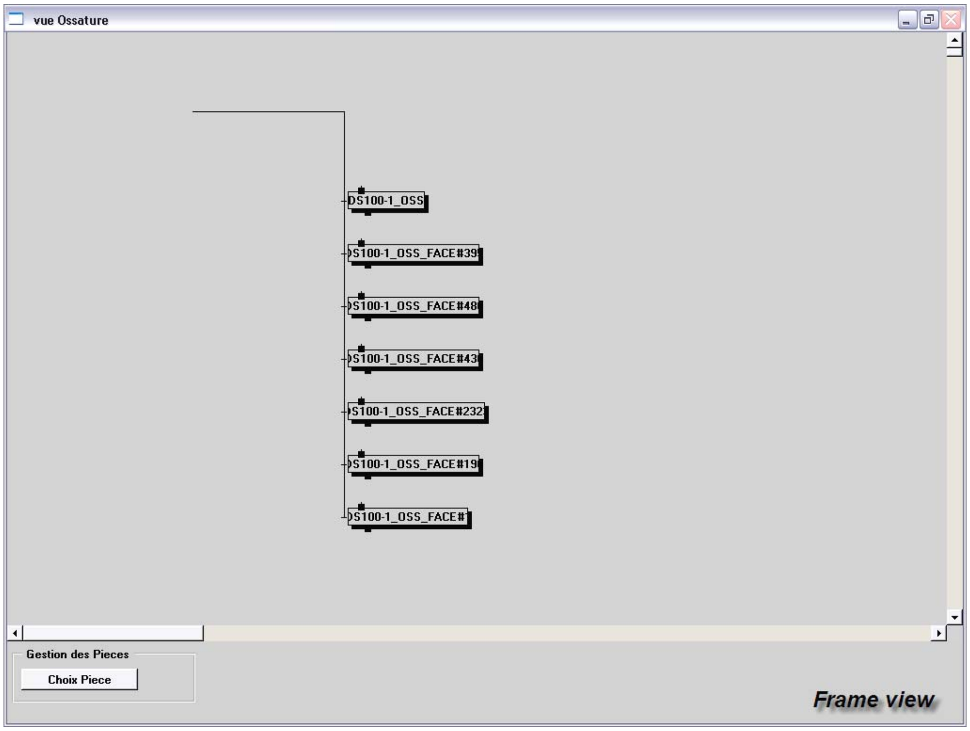


Figure 7.7 Initial information represented in frame view

Figure 7.8 represents the geometric view with the graphical representation and their default values retrieved from the STEP file. Note that these values can be modified at any time if one can contribute a more appropriate value for the product.

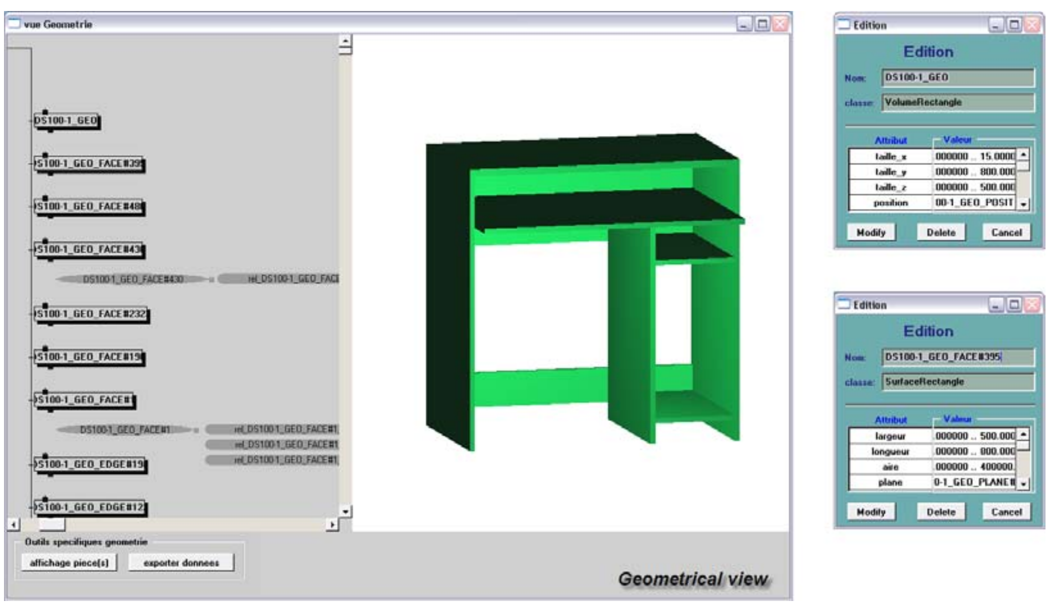


Figure 7.8 Initial information represented in geometric view

In the geometry view, the internal actor can automatically establish the preliminary geometrical constraint features that affect the function requirements of the product. This initial constraint features enhance the design actors to recognize the product structure – which part contacts with which part in which surface, which parts are parallel or perpendicular to each other, which parts are symmetry, etc. The data translation method has been developed to recognize such constraint features.

In order to recognize geometrical constraint features between the parts, we employ the internal actor to verify the coordination between the parts in the STEP file. Then, it defines the constraint features which identify the contact between surfaces of the parts in direction of axis-X, axis-Y, and axis-Z. Figure 7.9 represents, for example, the possibilities for defining constraint features between two parts. With these definitions, the internal actor creates automatically links and relation to establish such geometrical constraint features for the parts.

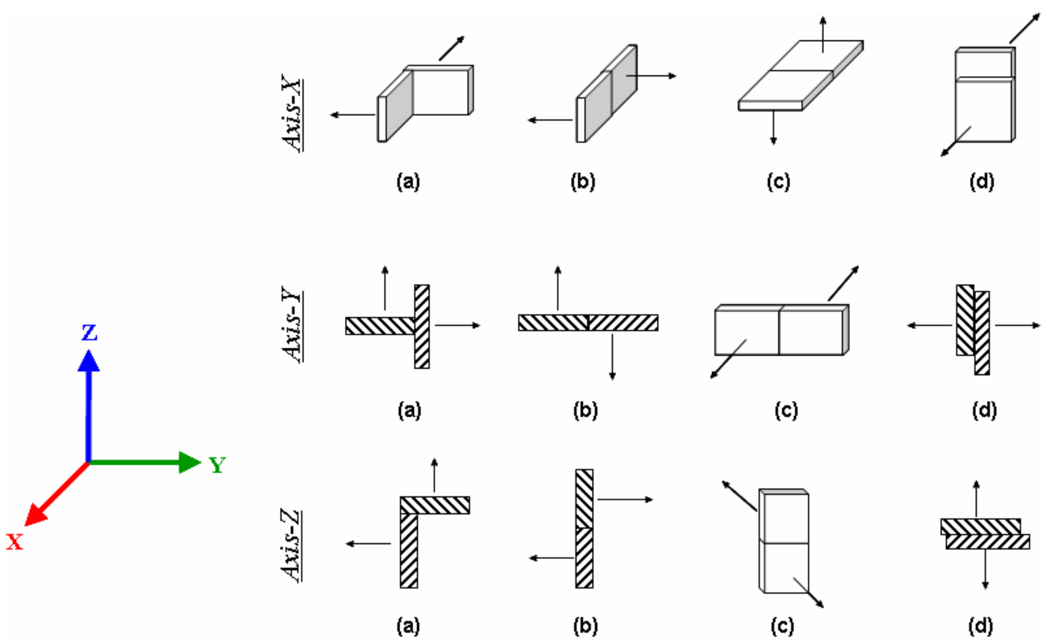


Figure 7.9 Example of geometrical constraint features

Such constraint features provide a context of the problem. To solve geometrical constraint problems, a solver must provide an instance of the given topology that exactly satisfies the given constraints [Shpitalni and Lipson 1997]. Let us consider by example, a definition of part *DS100-1_GEO* (the left part) and part *DS100-2_GEO* (the top part) as represented in Figure 7.10 (a), which contact perpendicularly to each other in the direction of axis-X. Therefore, a constraint feature *perpendicular* is applied to the surface of their parts, *DS100-1_GEO_FACE#430* and *DS100-2_GEO_FACE#*526. To define such constraint, the system creates a relation which is connected via a link of those two parts, with characteristic *plane*, as represented in Figure 7.10 (b) and Figure 7.11.

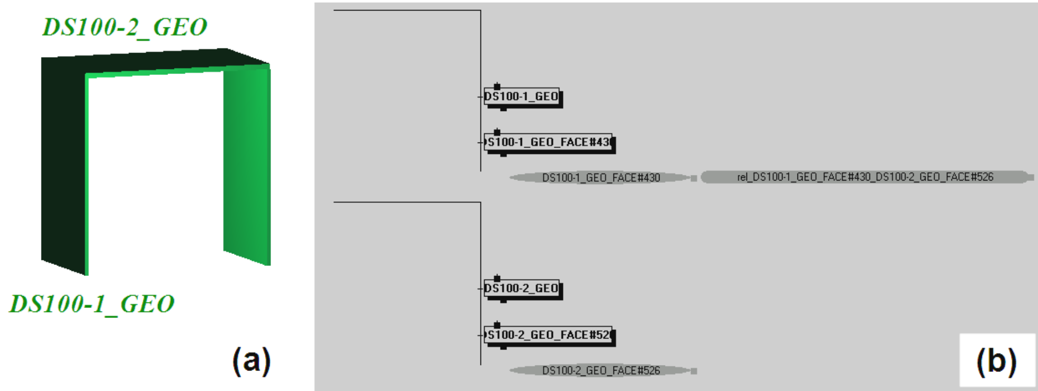


Figure 7.10 Example of a constraint feature in the geometric view

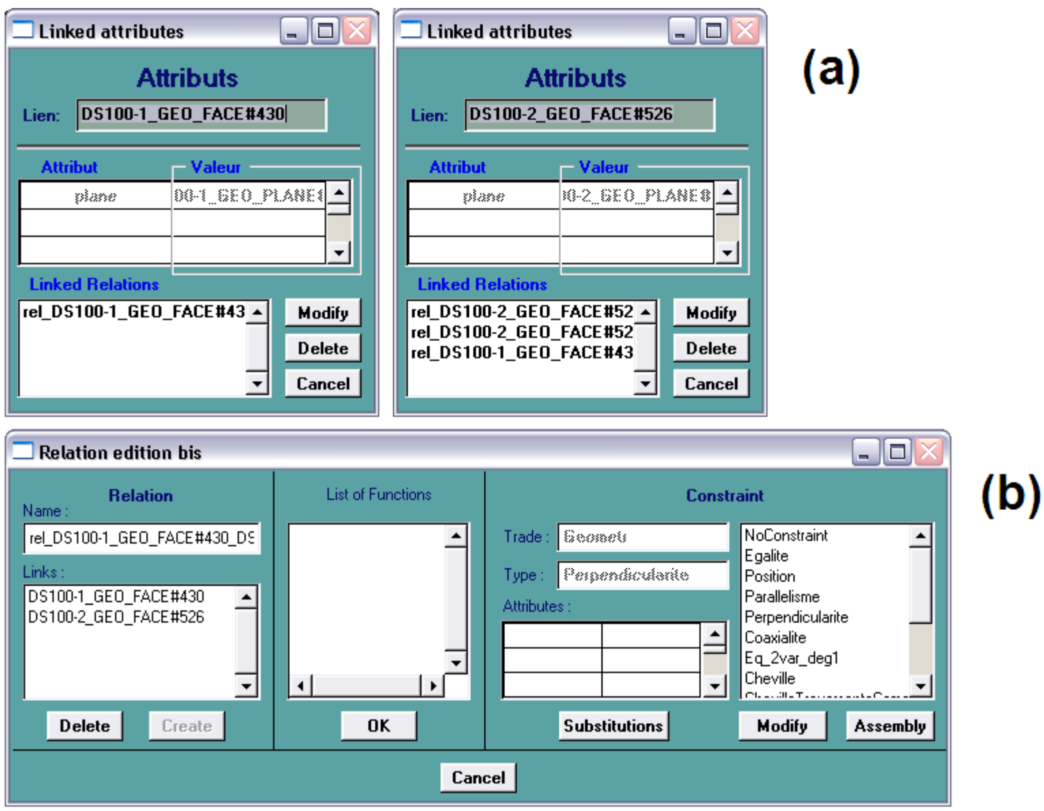


Figure 7.11 Panel represents constraint features in the geometric view

Note that one component can have more than one link and relation. It means that one part may have several/different constraint features. From these views, we can see that the internal actor facilitates the design team by creating automatically some data model (components, links, and relations) and features (both descriptive and constraint), which enhance the design actors to further evaluate the design.

Actually, the internal actor translates the STEP file not only to the technological view and those common views, but it also translates such data to the corresponding trade views. As presented in Chapter 6, this study takes into account three principal trades i.e. assembly view, mechanical view, and manufacturing view. Then the initiation phase of the three trade views is presented in the following sections:

7.3.1 Initiation of assembly view

At the first time when the assembler connected to the formal network and accessed to the shared database, there was no information in this trade view since it was a new project. However, during the translation process, the internal actor creates some initial data i.e. data model, entities, and empty features (relations without constraint) into this trade view. The assembly view comprises a basic representation and a graphical representation, as represented in Figure 7.12. The graphical representation facilitates the assembler to visualize the graphical model of the product as same as in the geometric view but it represents only the chosen parts. This facilitates the assembler to focus only the parts he interests to. Note that the assembler (and other trades) also has the common views as same as the technological view.

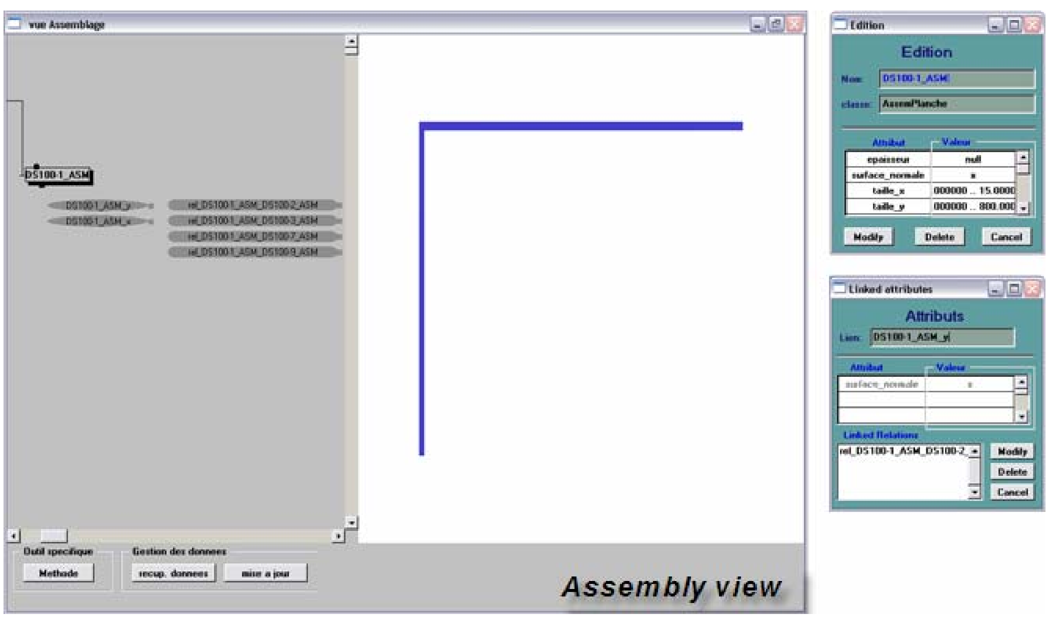


Figure 7.12 Initial information represented in assembly view

To permit the assembler to define constraint features (assembly solutions) for the product, the internal actor translates the constraint features in the geometric view, which concern to the assembler and then creates links and relations into the assembly view. Defining a constraint feature in the assembly view means to define a fastener for fastening between two parts. The created relations in this view at this time are empty constraint feature (*NoConstraint*). However, the system provides a specific panel that allows the assembler to choose an assembly solution, as represented in Figure 7.13 (a) and (b). The list of possible assembly solutions depends on the case of assembly given by Figure 7.9. This guides the assembler to determine an appropriate assembly solution for each pair of parts.

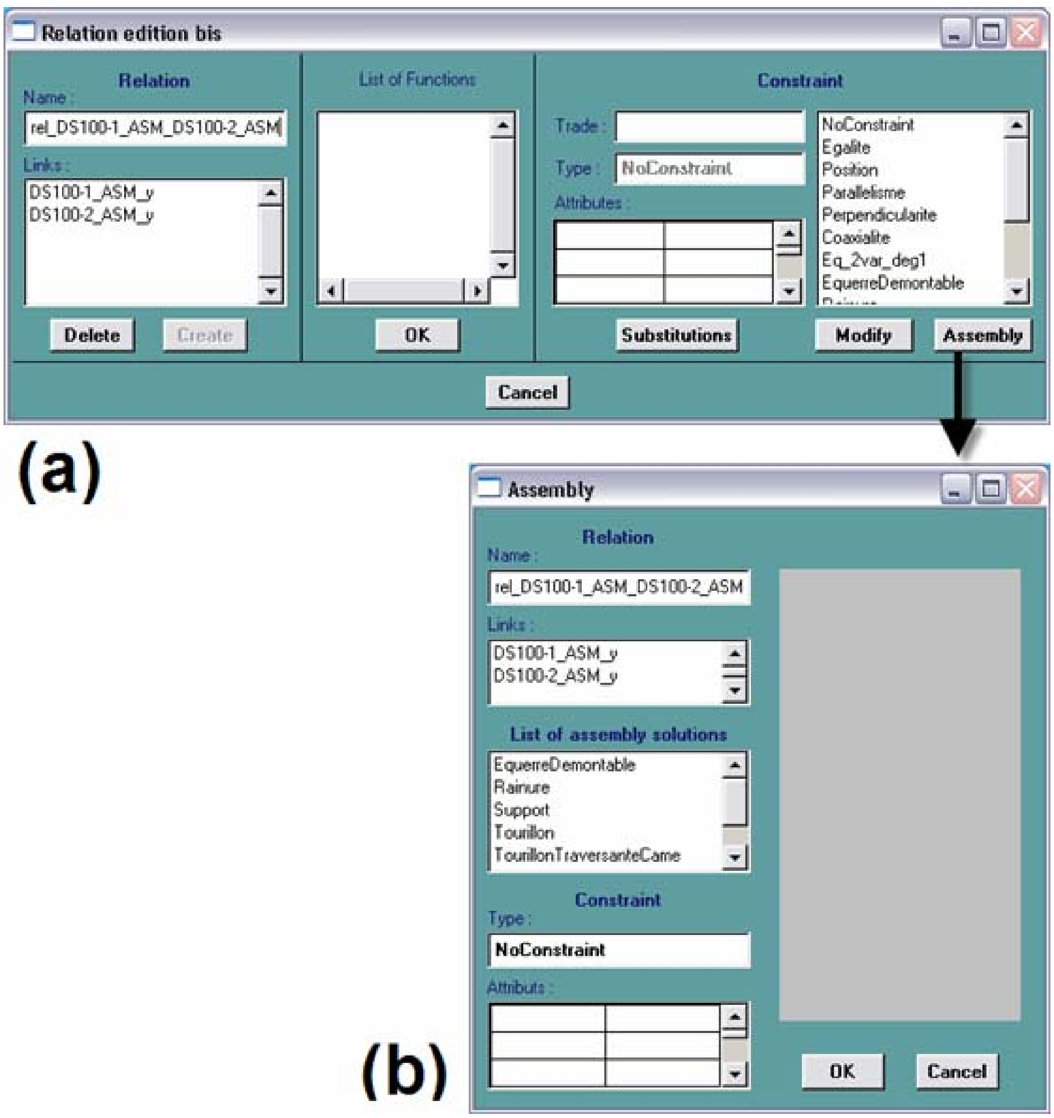


Figure 7.13 Panels represent constraint features in the assembly view

7.3.2 Initiation of mechanical view

This trade view concerns the quality aspect of the product. The mechanical view is required to evaluate stability, durability, strength of the product by testing the deflection of the parts. To contribute any information or to evaluate the design, the mechanician requires some initial data. As same as the assembly view, during the translation process of the STEP file, the internal actor creates some initial data, which support the mechanician to evaluate the design, into this trade view as represented in Figure 7.14.

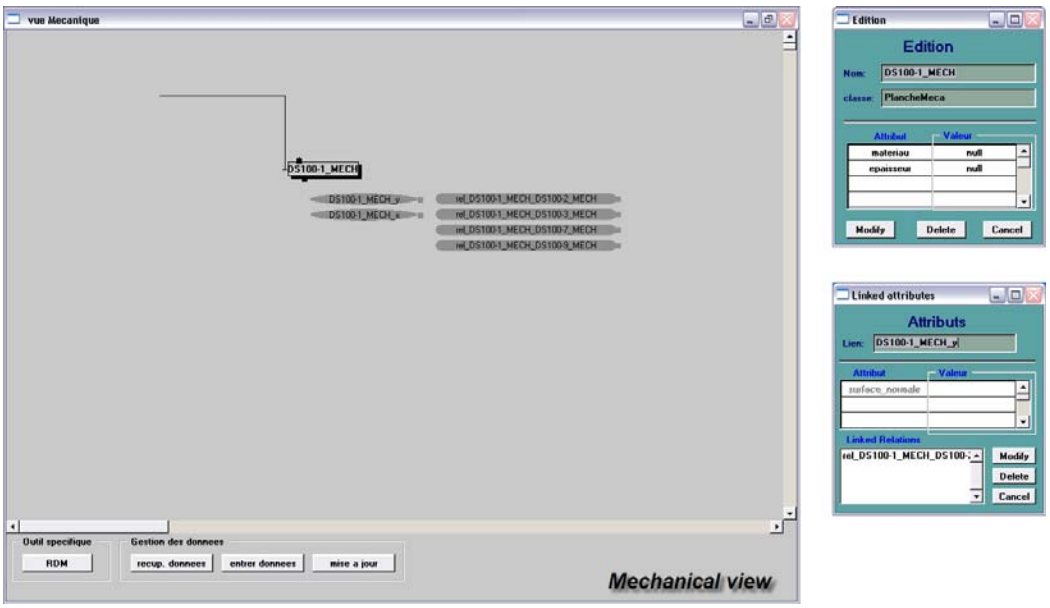


Figure 7.14 Initial information represented in mechanical view

7.3.3 Initiation of manufacturing view

This trade view has a major role in the design process. It concerns the evaluation of manufacturability, process planning, and production cost. As same as other trade views, the internal actor creates some data model, entities and features into this trade view to support the results of the evaluation, as represented in Figure 7.15. Characteristics of the created features that must be evaluated for example setup cost, operation cost, raw material cost, purchased parts cost and total cost of the product. However, these characteristics have not been defined yet until the evaluation has been done.

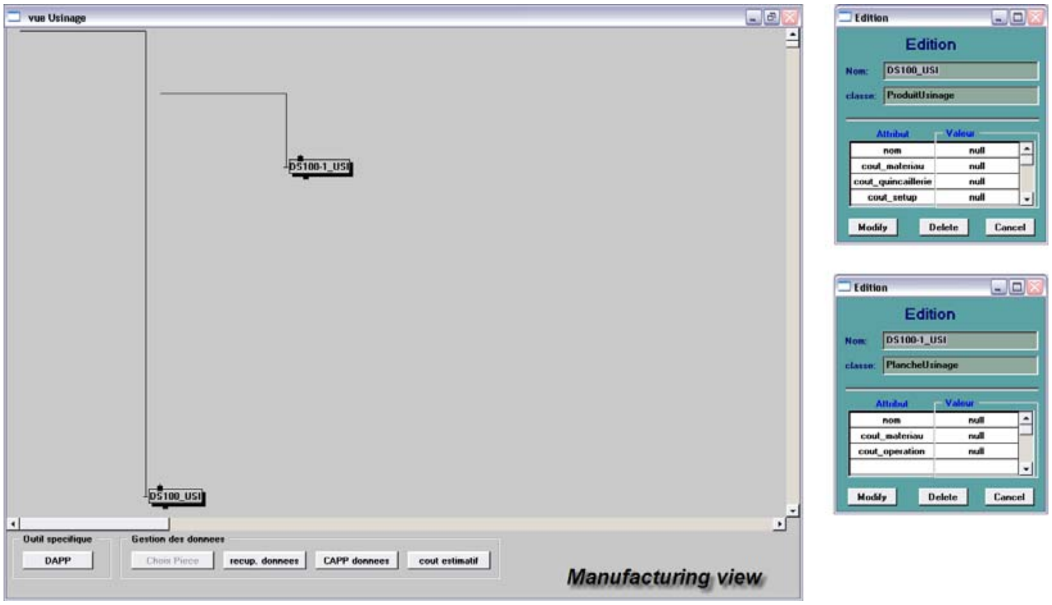


Figure 7.15 Initial information represented in manufacturing view

7.4 Demonstration of detail design phase

Following the initial design phase, this section presents the collaboration of the design team during the design process and the knowledge integration. Each design actor occupies on different tasks but they must contribute step by step their information, constraints, and points of view to the product through both the formal and informal communication network. CoDeMo provides GUIs, multi-view and multi-representation that facilitate the design actors to manipulate such information from/to the shared database.

7.4.1 Detail design of assembly view

The assembler concerns to define the assembly solutions to the product. The initial information in the geometric view enhances the assembler to visualize the overview of the product and its information. This trade view facilitates the assembler to concentrate on the parts which s/he is interested to. Since there is not only one possible assembly solution between two parts, CoDeMo creates in this trade view a library of fasteners that stores the choices of assembly, as represented in Figure 7.16.

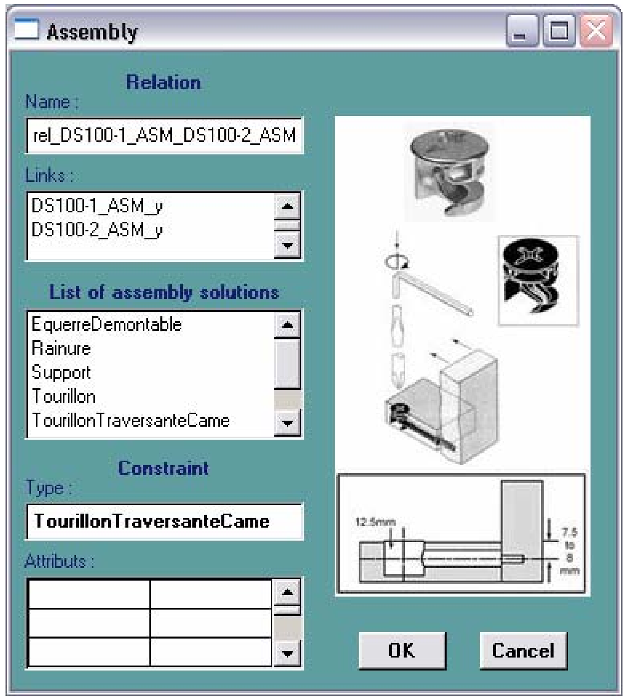


Figure 7.16 Assembly panel represents a library of assembly solutions

Furthermore, this assembly panel manipulates production rules into the library (see Chapter 5). It constrains the assembler to choose only the possible solutions for the parts. For example, to fasten between the part *DS100-1* (the left-side part) and the part *DS100-2* (the top part), as represented in Figure 7.17, the assembler can not apply a *support* to constrain these two parts due to the constraint in the geometric view; the part *DS100-2* is laid on the top of the part *DS100-1*. Otherwise, if it is chosen by mistake, the system will notify the assembler that it is a violated constraint, as shown by example in Figure 7.18.

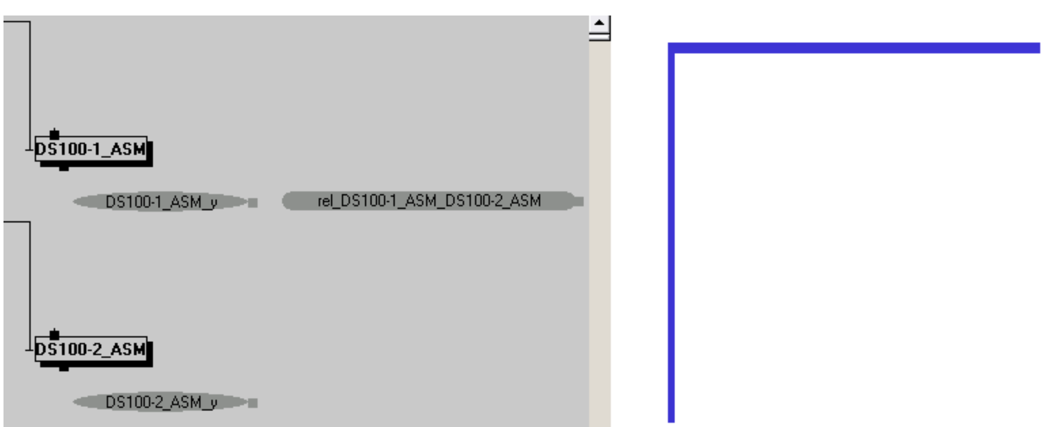


Figure 7.17 Example of assemble parts

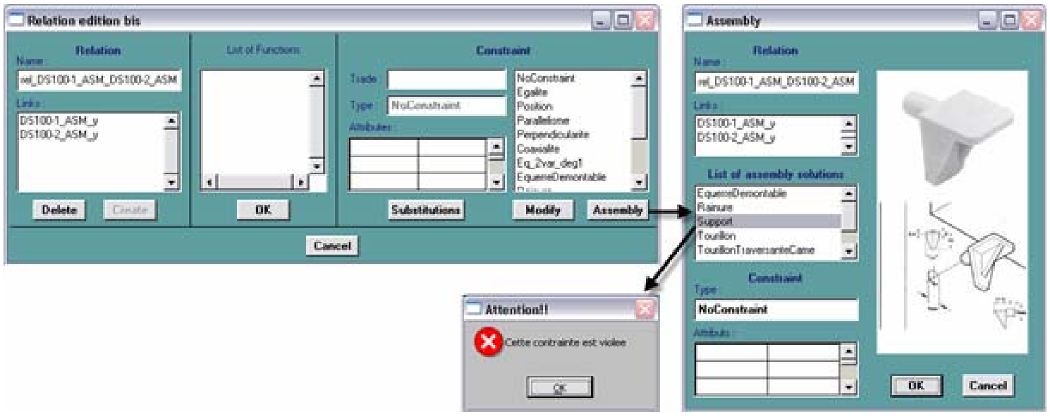


Figure 7.18 Example of a violated constraint

According to the contribution of the system and the initial information, the assembler can define assembly solutions (DPs) to the corresponding FRs of the design by manipulating such information together with competences and experiences in his/her domain. However, to define the details of such assembly solutions (values of the DPs) i.e. diameter, size, type of the fasteners and quantity, the assembler requires the more precise information such as thickness of the parts and/or the type of materials, which normally contributed by the mechanician. Furthermore, the chosen assembly solutions may constrain the manufacturing process in the manufacturing view. For that reason, the information of the design actors must be mapped to each other. This mapping process is driven by the data translation and data propagation method (see 4.3.8). The notification function in the mapping process notifies the corresponding actors to recognize the established DPs created by other actors. We present an example of mapping process. Following the example in Figure 7.17, the assembler may define a constraint *TourillonTraversanteCame* to fasten those two parts, as represented in Figure 7.19.

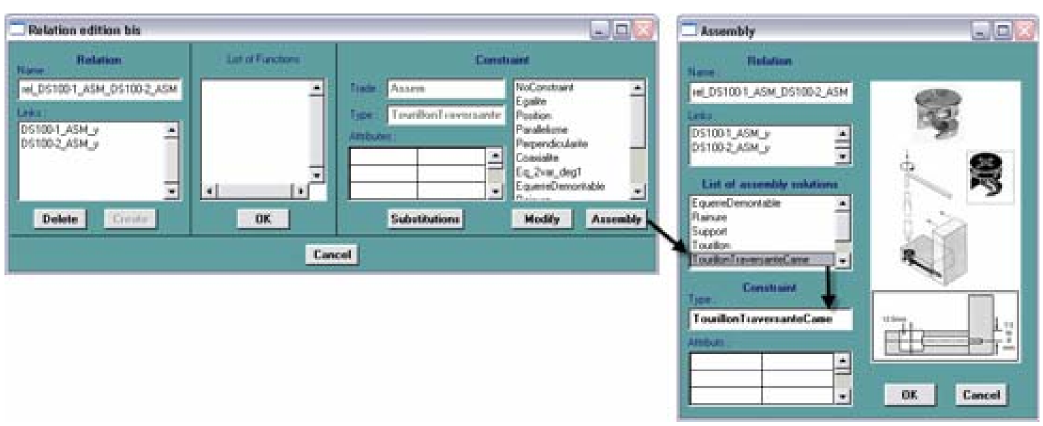


Figure 7.19 Example of choosing a constraint in assembly view

This assembly solution uses a set of knock-down fitting that consists of a connecting bolt and housing. This choice constrains the manufacturer to drill the part *DS100-1* two holes. One vertical hole is for supporting the housing and the other horizontal one is for the connecting bolt. To notify the manufacturer to perceive the manufacturing process of the parts, [Radulescu 2005] developed a neutral file named '*QTrans*' for associating to the translation process (see 4.3.8). This *QTrans* file facilitates the internal actor to translate such constraint features. The description of the structure of *QTrans* file can be found in Chapter 5. Table 7.2 shows an example a knowledge module presented in QTrans file.

Table 7.2 Example of knowledge module presented in QTrans file

```
Component_Name
        TourillonTraversanteCame Assem name

Traduction
        Component Percer Usinage name_1_USI
        Component Percer Usinage name_2_USI
        Link name diametre_boitier name_diametre_boitier
        Link name epaisseur_boitier name_epaisseur_boitier
        Link name diametre_boulon name_diametre_boulon
        Link name longueur_boulon name_longueur_boulon
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Link name_2_USI diametre name_2_USI_diametre
        Link name_2_USI epaisseur name_2_USI_epaisseur
        Relation name_diametre_boitier  name name_1_USI_diametre name_1_USI
        relation_name_1
        Relation name_epaisseur_boitier name name_1_USI_epaisseur name_1_USI
        relation_name_2
        Relation name_diametre_boulon name name_2_USI_diametre name_2_USI
        relation_name_3
        Relation name_longueur_boulon name name_2_USI_epaisseur name_2_USI
        relation_name_4
@
```

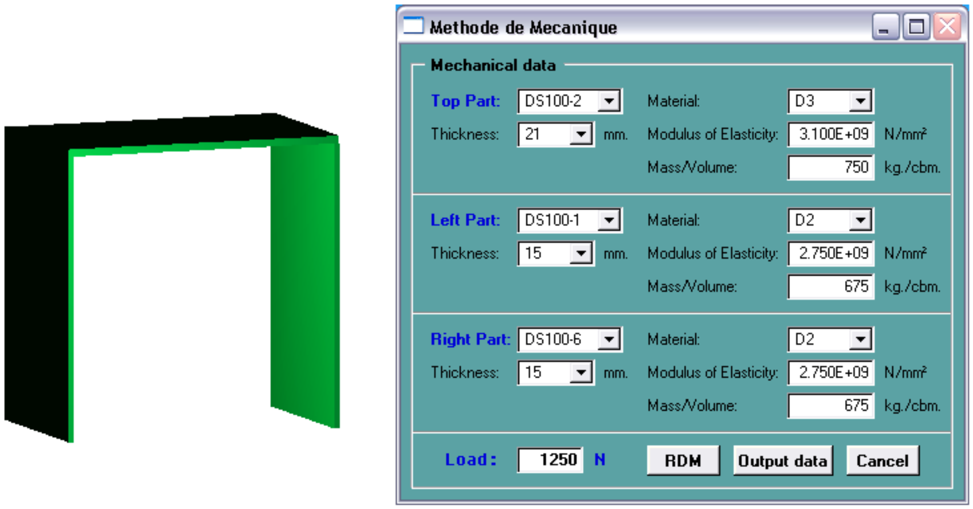
As soon as the assembler has created the constraint feature *TourillonTraversanteCame* or any, the internal actor maps this feature to the shared database. It consequently translates such feature by using the production rules, and then propagates a new feature(s) to the manufacturing view. As represented in Figure 7.20, the feature *TourillonTraversanteCame* contains the characteristics of the fastener such as diameter of connecting bolt and housing, size and quantity while the translated features *Percer* in the manufacturing view contain the characteristics of process such diameter and depth of hole. Those created features (DPs) may contain empty values due to unknown or insufficient of information, however such values can be realized as soon as the design actors have enough information. This method permits the design actors to work on the fuzzy problem.

Figure 7.20 Example of assembly feature translation and propagation

7.4.2 Detail design of mechanical view

One of the most important issues that the customer may concern is the quality of the product. The objective of this mechanical view is to guarantee the stability and durability of the product. CoDeMo creates in this trade view a library for storing information of available materials, which supports the mechanician to run a deflection test, e.g. material types, physical and mechanical properties of materials – density, modulus of rupture, modulus of elasticity, etc. The task of this trade view is to define the appropriate type of materials and the thickness of the parts.

The deflection of the plates depends on the structure of the product and a given load respecting to the standard's requirement. To run such mechanical test, CoDeMo employs a tool named *RDM6* [Debard 2000a, Debard 2000b] as a specific application of the mechanician. *RDM6* is developed by Yves Debard of the Institut Universitaire de Technologie du Mans. The objective of this tool is to calculate the structures by using the finite elements method. We apply this tool in this study to estimate the deflection of the parts. We may consider by example the main structure of the product which comprises of the left-side part, the top part, and the right-side part, as represented in Figure 7.21 (a). CoDeMo creates a panel that facilitates the mechanician to contribute the information as represented in Figure 7.21 (b).



(a) (b)

Figure 7.21 Panel of input mechanical data

This panel is linked to the library that stores information of the available materials. It displays the materials by thickness and type of materials while the properties of the chosen materials will be automatically presented. At this time, the mechanician has to define the information to perform the mechanical test. Suppose that the given load for testing is 1250 N. As soon as the mechanician has completed the required information, the internal actor will be asked to translate and to output this information in order to be used for evaluating the design with *RDM 6*. Note that before defining the parameters of the parts, the mechanician usually sets the priority of materials respecting to the customer's requirements such as cost, aesthetic, or quality. It means that if the product concerns first the cost, the mechanician chooses the most appropriate materials sorting by its cost before running the test.

As presented before, it is necessary to understand the exchanged format of the specific trade applications in order to perform the integrated design. The internal actor is then developed to realize the data format of *RDM 6*. This function facilitates the system to exchange information between CoDeMo and the specific trade applications. As a result, the mechanician can output the information by click on the **Output data** button on the panel. Table 7.3 represents a part of information of output file. The user can run *RDM 6* via the panel by click on the **RDM** button and then chooses the exported file to open in the specific application *RDM 6*. Figure 7.22 presents a mechanical model of the example in Figure 7.21 after choosing its exported file.

Table 7.3 Example of exported file format

```
////
$noeuds ( 4 )
   1  0.00000000000E+00  0.00000000000E+00  0.00000000000E+00
   2  0.00000000000E+00  8.00000000000E-01  0.00000000000E+00
   3  8.00000000000E-01  0.00000000000E+00  0.00000000000E+00
   4  8.00000000000E-01  8.00000000000E-01  0.00000000000E+00
   0
$poutres ( 3 )
   1 RIRI     1    2  0.00000000000E+00 -0.00000000000E+00  1.00000000000E+00  2  2
   2 RIRI     3    4  0.00000000000E+00 -0.00000000000E+00  1.00000000000E+00 10 10
   3 RIRI     2    4  0.00000000000E+00  0.00000000000E+00  1.00000000000E+00 17 17
   0
$SECTIONS
2
TYPE PARAMETREE
NOM *Rectangle plein
DESIGNATION *LY = 15 LZ = 500.000000 mm
LOGO 5
DIMENSIONS 2
 1.500000E-02
 5.000000E-01
AIRE  7.50000000000E-03
IYY  1.56250000000E-04
IZZ  1.40625000000E-07
WPY  9.37500000000E-04
WPZ  2.81250000000E-05
TORSION  5.51868750711E-07
KYY  1.0000000
KZZ  1.0000000
IWW  0.00000000000E+00
///
10
TYPE PARAMETREE
NOM *Rectangle plein
DESIGNATION *LY = 15 LZ = 500.000000 mm
LOGO 5
DIMENSIONS 2
 1.500000E-02
 5.000000E-01
AIRE  7.50000000000E-03
IYY  1.56250000000E-04
IZZ  1.40625000000E-07
WPY  9.37500000000E-04
WPZ  2.81250000000E-05
TORSION  5.51868750711E-07
KYY  1.0000000
KZZ  1.0000000
IWW  0.00000000000E+00
///
```

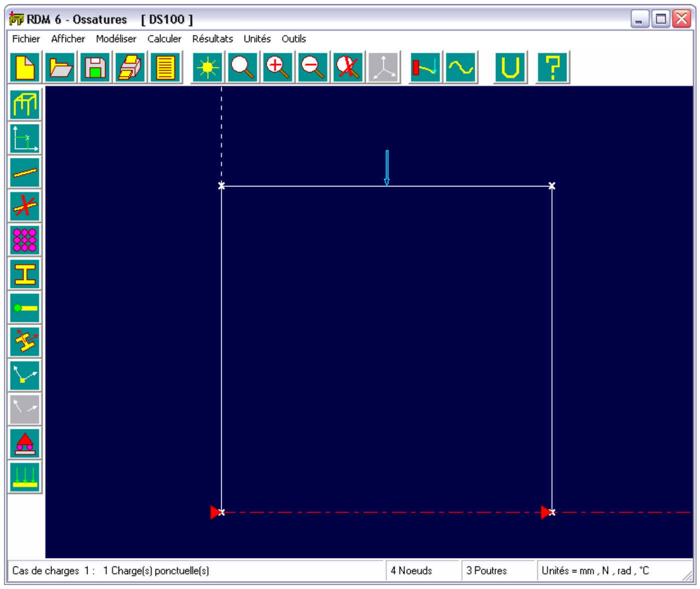


Figure 7.22 Example of a mechanical model represented in RDM 6

The output file is imported into *RDM 6* and presents the information as same as presented in CoDeMo as shown in Figure 7.23. Nevertheless, the mechanician may regulate or add some parameters if needed respecting to the standard rules. For example, the standard may require fixing the table's legs during the test. We simulate a deflection test of this example with a given load 1250 N. based on the imported data from the mechanical view and also fix the table's legs to the ground. *RDM 6* gives us the result of simulation as represented in Figure 7.24. The maximum value of deflection for the left-side and the right-side part is 3.007 mm. at the point 528 mm from the fix point (ground) while the maximum value of deflection for the top part is 7.888 mm. at the middle point 400 mm. The results of the deflection test of these parts are represented as diagram as shown in Figure 7.25 from top to down respectively.

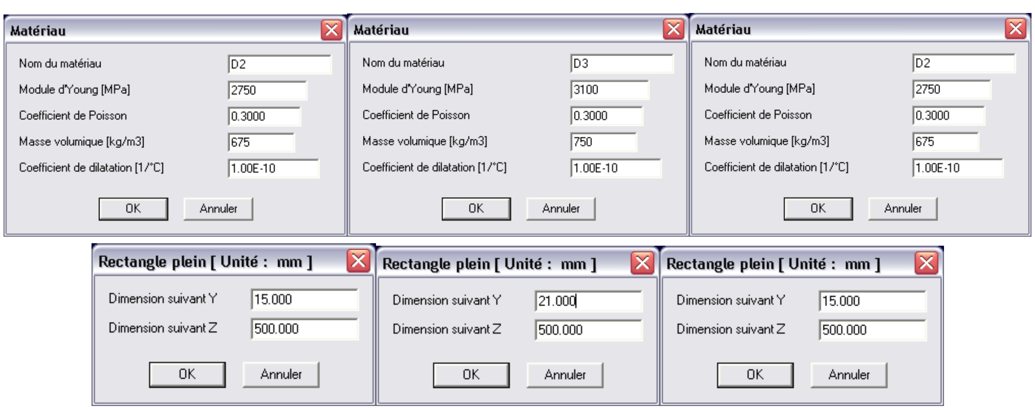


Figure 7.23 Information of a mechanical model represented in RDM 6

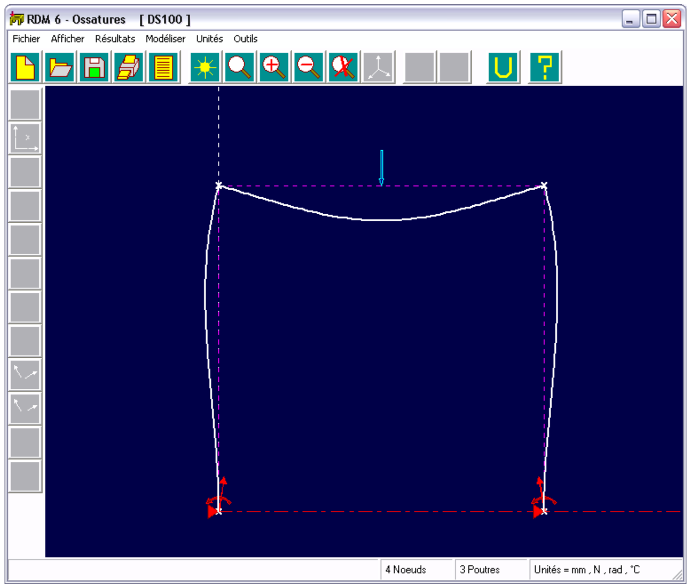


Figure 7.24 Simulation of deflection testing

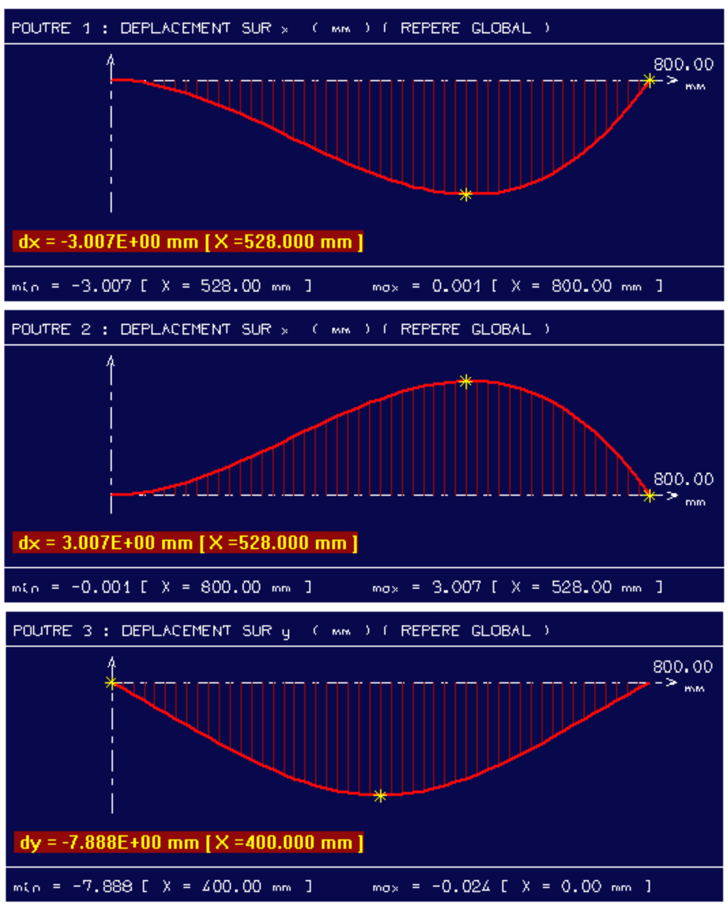


Figure 7.25 Results of deflection testing

Note that this test is just to estimate the deflection of the parts. Otherwise, to precise the result obtained from the real physical test, it may require more parameters and may be more complicated. Furthermore, it also depends on the methods and solutions of the standard's requirements. With these results, the mechanician may accept or refuse the chosen parameters respecting to the standard's requirements. If the results are not accepted, the mechanician has to change the parameters/values regarding to the appropriate priority. S/he can input directly the new parameters/values by the interface panels as shown by example in Figure 7.23. This process will be achieved when the mechanician obtains the most appropriate values and parameters. As soon as the process is done, the mechanician must notify the design team the new information by providing the results to the shared database. CoDeMo allows the mechanician to create a text file that stores the results of the deflection test as represented in Table 7.4. In order to trade in the results to the shared database, we develop the *QTrans* file with a new format of knowledge module as represented in Table 7.5. This knowledge module transforms the results from the mechanician into a format which the internal actor is familiar with. The description of this new format of knowledge module can be found in Chapter 5.

Table 7.4 Example of results from mechanical view

```
Materiaux
  DS100-1 D2
  DS100-2 D3
  DS100-3 D2
  DS100-4 D2
  DS100-5 D2
  DS100-6 D2
  DS100-7 D2
  DS100-8 D2
  DS100-9 D2
@
Epaisseurs
  DS100-1 16
  DS100-2 19
  DS100-3 15
  DS100-4 16
  DS100-5 15
  DS100-6 16
  DS100-7 15
  DS100-8 15
  DS100-9 3
@
```

Table 7.5 Example of knowledge module between mechanical and assembly view

```
Component_Attribute
        PlancheMeca Meca name_MECH
        Tourillon Assem name_ASM

Traduction
        Attribute name_MECH materiau Char materiau_planche
        Attribute name_MECH epaisseur Float epaisseur_planche
        Attribute name_ASM epaisseur Float epaisseur_planche
        Link name_MECH epaisseur name_MECH_epaisseur
        Link name_ASM epaisseur name_ASM_epaisseur
        Relation name_MECH_epaisseur name_MECH name_ASM_epaisseur name_ASM
        relation_MECH_1
@
```

At this time, the mechanician asks the internal actor to map this information to the shared database, to translate and to propagate such information to the corresponding views. To begin the translation process, the mechanician clicks on the mise a jour button from the specific tool panel. This will display a panel with a list of parts that allows the mechanician choosing the part to add/update data. The mechanician can update any part as soon as s/he perceives its values without waiting all the evaluation process has done. We present in this section the feature translation and propagation from the mechanical view to the assembly view. Once the assembler chooses a part from the *Part Selection* panel, the internal actor is asked to map this information to the shared database and to propagate to the corresponding views as represented by example in Figure 7.26 how to update the result of the part *DS100-1_MECH* in the mechanical view to the part *DS100-1_ASM* in the assembly view.

Due to the feature translation and propagation in the mechanical view, the assembler can consequently define the characteristics' values of the chosen assembly solutions. Following the example in Figure 7.20, the assembler has chosen *TourillonTraversanteCame* as a constraint feature to fasten between the part *DS100-1* and the part *DS100-2*. We may present continually in this section the consequences of this example. From Figure 7.26, the mechanician defines the type of material of the part *DS100-1* as 'D2' which is one type of particleboard and the thickness is 16 mm while the type of material of the part *DS100-1* is 'D3' and the thickness is 19 mm. Regarding to this constraint, the assembler has to choose the most appropriate of a set of knock-down fitting that fits to the type of material and the thickness of those parts. Following to the given constraint, suppose that the assembler chooses a set of knock-down fitting which has characteristics as following: the housing with diameter 12 mm. and 10 mm. of thickness, a connecting bolt with diameter 8 mm. and 48 mm. long. To realize such constraint and knowledge, the mechanician and the assembler have to create production rules to translate characteristics of such features. The created production rules will be stored in as a module of knowledge in *QTrans* file. Table 7.6 shows by example a knowledge module between assembly and manufacturing view.

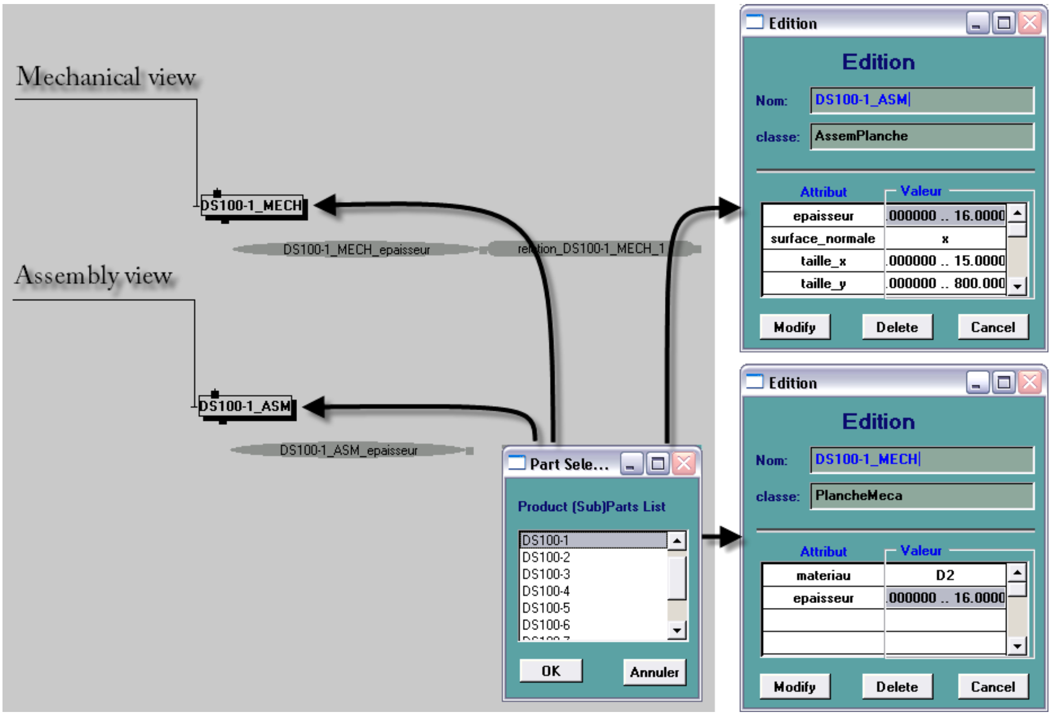


Figure 7.26 Example of mechanical feature translation and propagation

To illustrate the knowledge translation and propagation in the manufacturing view, we present continually the example of the part *DS100-1*. As same as the mechanician, the assembler begins the translation process by click on the 

button from the specific tool panel. The *Part Selection* lists the parts and allows the assembler to choose. As soon as s/he has chosen a part, in this case, the part *DS100-1* is chosen, the internal actor is asked to translate the module of knowledge, as represented for example in Table 7.6, and to propagate such information (add new characteristic of features) to the corresponding (sub)components in the assembly view and manufacturing view. Regarding to the knowledge module, created from the production rules, the result of the translation process is represented in Figure 7.27.

Table 7.6 Example of knowledge module between assembly and manufacturing view

Component_Name
    Tourillon Assem name

SubComponent_Name
    SubComponent Percer Usinage name_1_USI
    SubComponent Percer Usinage name_2_USI

Traduction
    Attribute name type Char type_tourillon
    Attribute name diametre Float diametre_tourillon
    Attribute name longueur Float longueur_tourillon
    Attribute name quantity Int qty_tourillon
    Attribute name_1_USI diametre Float diametre1_tourillon
    Attribute name_1_USI epaisseur Float epaisseur1_tourillon
    Attribute name_2_USI diametre Float diametre2_tourillon
    Attribute name_2_USI epaisseur Float epaisseur2_tourillon
@

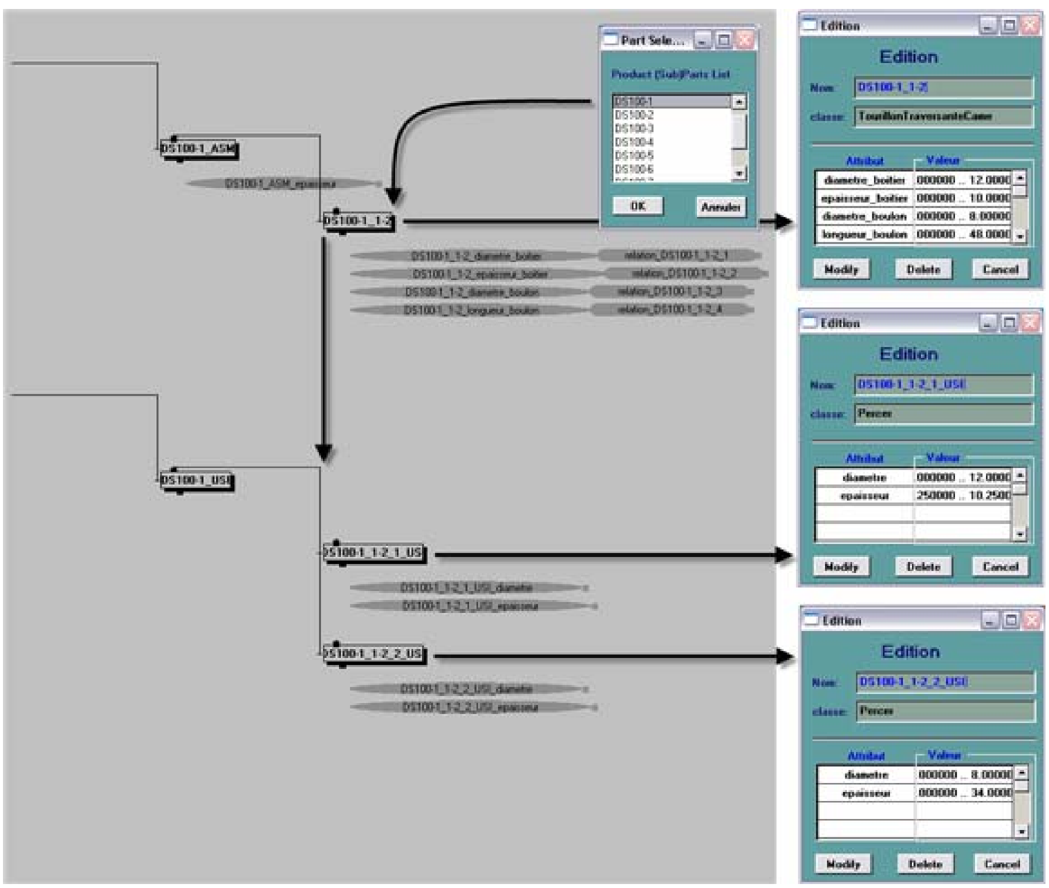


Figure 7.27 Example of knowledge translation and propagation

7.4.3 Detail design of manufacturing view

The cost of the product is the most important issue in the final decision process. It reflects the margin profit of the product and the potential of competition of the company. Thus, it is essential to estimate the cost and the manufacturability of the product at the early stage as soon as possible. To do so, the manufacturer is required to gather the corresponding information as much as possible. The initial data and the information which are contributed from the previous trade views are important. Such information enhances the manufacturer to establish DPs and its values afterward e.g. chosen assembly solutions, type of materials, diameter and depth of a hole, width and depth of a groove, etc.

In fact, there is some information in the common views that needs the design actors who are in charge of, to define during the design process. For example, the tolerance of the parts which may be defined by the assembler or the manufacturer; the materials using to cover the edges and the surfaces of parts (paper, melamine, PVC, veneer), and its characteristics (color, thickness), which may be concern by the sketcher, etc. Figure 7.28 represents by example *edition panel* in the frame view for contributing such information. Such information also supports the manufacturer to determine the process planning and to estimate the product cost.

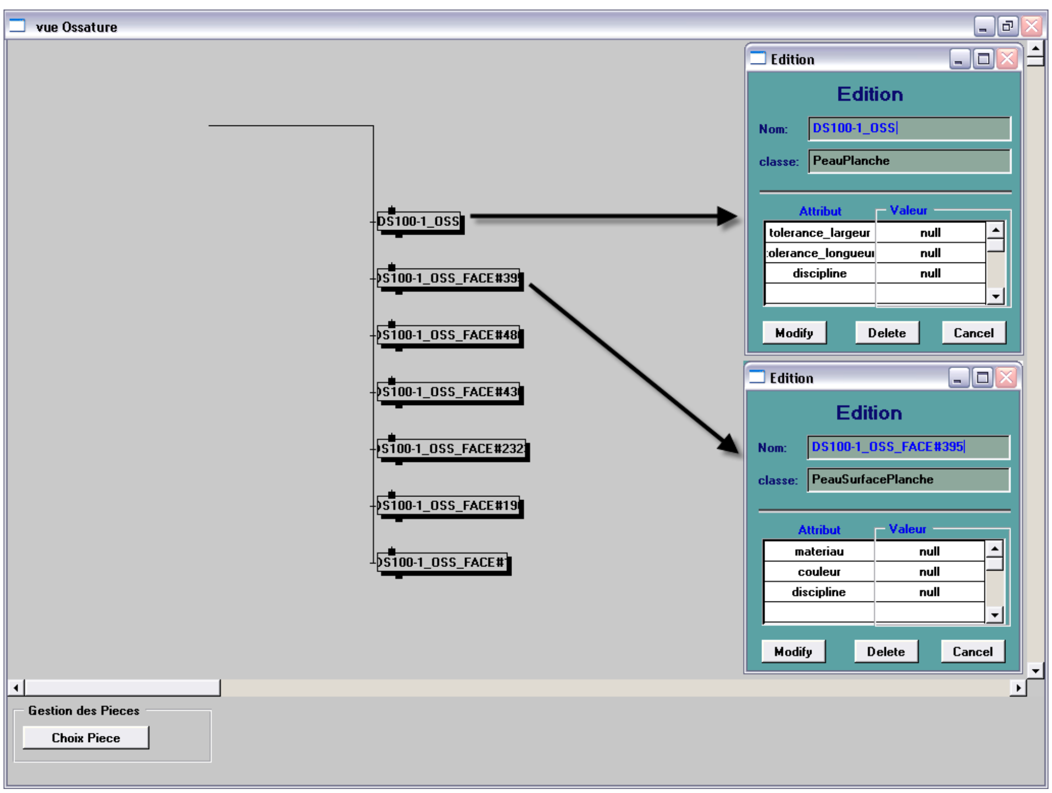


Figure 7.28 Edition panels in the frame view

However, to estimate the cost and to plan the manufacturing process, the manufacturer requires further a lot of information. Due to the requirements of numerous information of the manufacturing system, only the information contributed by the design team is not sufficient. The manufacturer requires an assist tool to evaluate the design more than the existing facilities. Therefore, in this study, we develop a specific application named *DAPP*, a Database Application for Production Planning. This application facilitates the manufacturer to manipulate the information contributed from CoDeMo into the manufacturing database.

As soon as the design team has contributed information as much as necessary, the manufacturer asks the internal actor to output the information by click on the 

button and then selects a part, in this case is the part *DS100-1*. The internal actor will retrieve the corresponding information and output it into a neutral file, as represented in Figure 7.29. Note that an example of this output file can be found in Annex II.

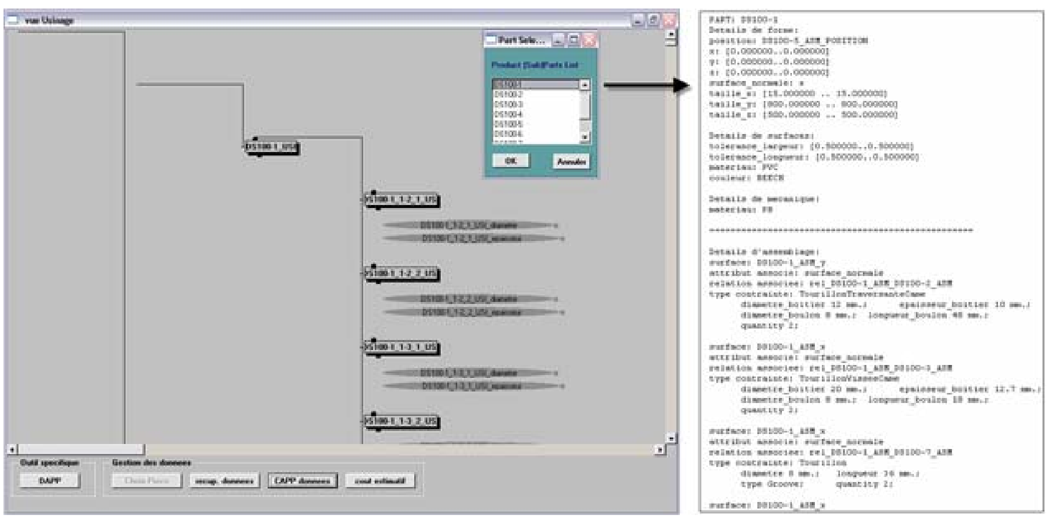
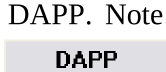


Figure 7.29 Example of an output file from CoDeMo

To utilize such information, we develop DAPP to be able recognize the output data from CoDeMo. The manufacturer uses DAPP to trade in the output data into the database of DAPP. Note that the manufacturer can run the specific application by click on the 

button in the specific tool panel. This information enhances the manufacturer to initiate the evaluation of manufacturing process. We present consequently the evaluation process and the application DAPP here after.

7.5 Specific trade application for manufacturing expert

This section presents a specific application using in wood furniture industry. It is developed to assist the manufacturer to evaluate the design. Wood furniture industry is one of the highest competitions in global manufacturing environment. The growth of wood furniture has been regularly increasing for a long time particularly the furniture made of particle board and medium-density fiber board. This sort of furniture has a short life cycle and rapid change of models and styles [Butdee 2002]. In addition, such product consists of various parts and a large number of information. Therefore, it is difficult and complicated to manage the manufacturing process only by an integrated design modeller. A literature review of the current status of Computer-Aided Design (CAD) and Computer-Aided Process Planning (CAPP) software technologies reveals the lack of interface standards to enable the integration of these systems [Feng and Song 2000]. Although the cost is principally incurred during the production process but the major cost of the product is committed in the design stage (see Figure 2.4). Therefore, Production constraints must be taken into account at the same time as economic, logistics or legislative constraints. Facts and constraints (knowledge model) must be structured, formalized and represented [Martin and D'Acunto 2003].

According to the study in wood furniture industry for years, we have found that one of the most fundamental problems of production is that they have not enough information for planning a good manufacturing process plan. Therefore, we develop a Database Application for Production Planning, *DAPP*, to facilitate the tasks of manufacturer or a person who is in charged of production planning. The main objective of this application is to support the manufacturer to create a conceptual process planning, and to estimate the manufacturing time and the production cost.

During the period of studying in the wood furniture companies, we had applied DAPP to collect necessary information and then manipulated it as modules of knowledge into the database of *DAPP*. We have succeeded in using *DAPP* with the wood furniture companies we have visited. Nevertheless, to apply *DAPP* for using in this study, we have to develop *DAPP* to be able to exchange information with *CoDeMo*. The manufacturer begins the evaluation by retrieving the pertinent information from CoDeMo, using the information in the database of *DAPP*, and then manipulates his/her knowledge into the database of *DAPP* for evaluating the design. To end the evaluation process, *DAPP* outputs the results in a form of knowledge module, which stores in a neutral file, and then manipulate such knowledge into the system. To achieve this task, we postulate that the user in the manufacturing view must be an expert in this domain. As a result, s/he can contribute and manipulate his/her knowledge and experience into *DAPP* for evaluating the design. In this section, we present the functions of DAPP and how the manufacturer uses it to evaluate the design since obtained the pertinent information from CoDeMo.

7.5.1 Structure of DAPP

DAPP has been developed by using MS Visual Basic to create the interface forms and using MS Access to create a relational database that linked to DAPP via *ODBC*[22] driver, as shown in Figure 7.30.

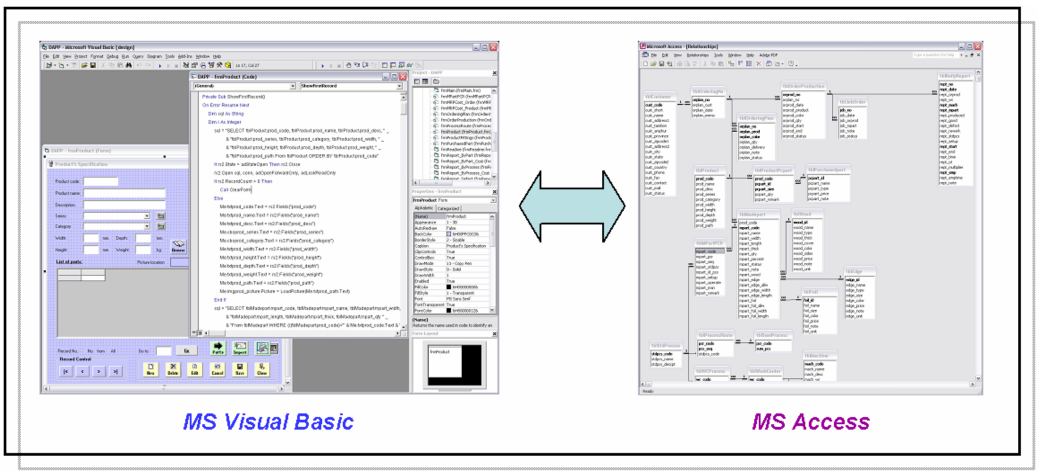


Figure 7.30 Tools for developing DAPP

Two principal objectives lead DAPP: one is to satisfy the development of design process by evaluating; the other is to satisfy the development of industrial part. To satisfy the development of design process, we have to evaluate the design both of new model and developed model by creating a conceptual process planning and estimate the manufacturing time and production cost. To satisfy the development of industrial part, DAPP monitors the manufacturing process and analyzes the results of the process planning. The monitoring and the analysis enhance the producer to develop the process planning being more accurate for the next time. Figure 7.31 shows the main interface of DAPP as a result of the development.

Although, this view concerns mainly the manufacturing section, it requires as well the information of other views in order to perform the evaluation. We realize this condition, so we develop a relational database to store the pertinent information from those views. The interfaces of DAPP comprise different sections. These interfaces facilitate the users to contribute their information and knowledge into the database. On the other hand, they also contribute the results of the evaluation. DAPP is

[22] Open Database Connectivity

composed of six sections. Each section contains some interfaces for input and output information as represented in the overview of the structure of DAPP, in Figure 7.32. To present the functions of DAPP, we divide the interfaces into two groups: *Input information* and *Output information*. We present first the *input information* that is the source of information, and then the *output information* that is the result of evaluation.

Figure 7.31 Main interface of DAPP

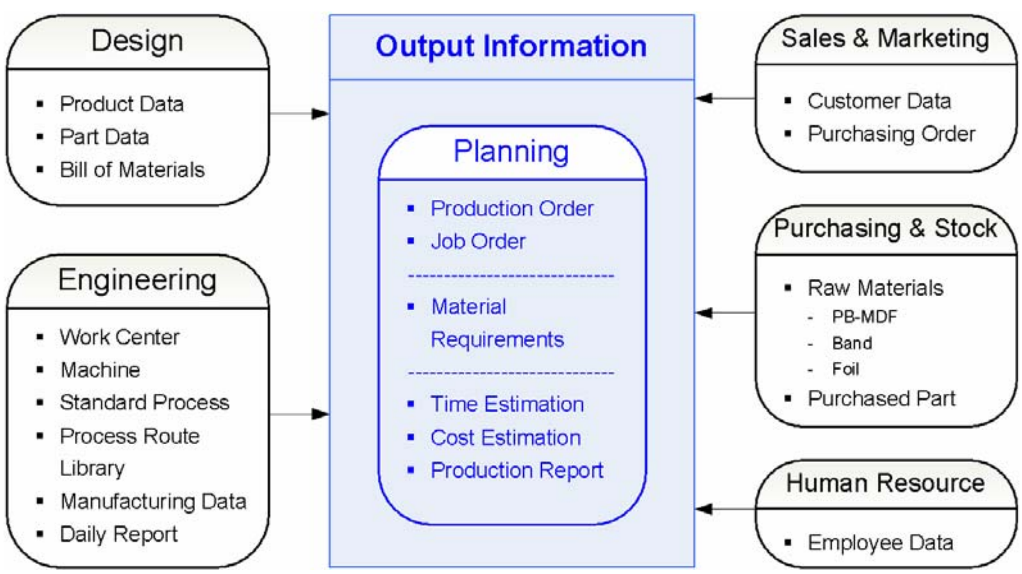


Figure 7.32 Structure of DAPP

7.5.2 Input information

Input information consists of five sections i.e. Design, Engineering, Sales & Marketing, Purchasing & Stock, and Human Resource. Each section presents the interfaces that facilitate the users to contribute their information and knowledge. Furthermore, it permits the users to access the information, to edit, to modify, to add and/or to delete the information.

Sales & Marketing

This section concerns the information about customer and purchasing order from the customer. It concerns the coordination with the customer since making contact with the customer, collection of information, negotiation, and confirmation until the deal is done. This section contains two interfaces i.e. *Customer Data* and *Purchasing Order*.

The interface of *Customer Data* stores the list and details about customers, as represented in Figure 7.33. This customer data is requested by the *Purchasing Order* that stores the details of proforma invoice e.g. description of ordered items, delivery date, as represented in Figure 7.34. The information of *Purchasing Order* is corresponding to the *Planning* section in order to create a controlled document such a *Production Order*, which will be presented in the section of *Output information*.

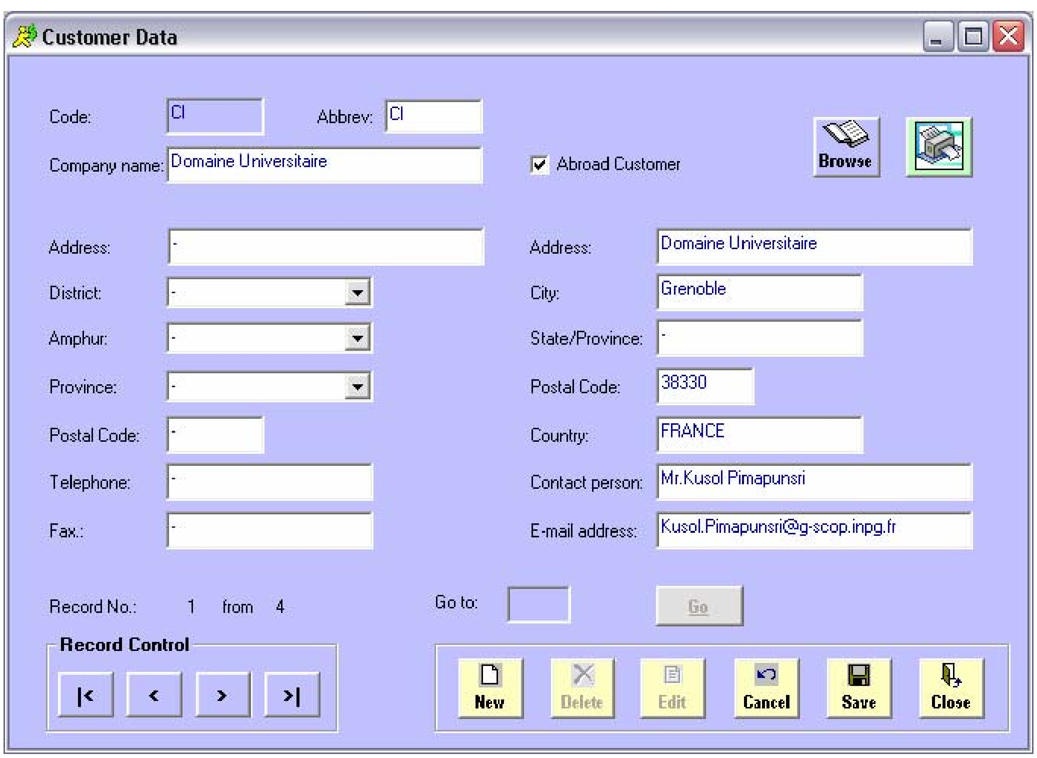


Figure 7.33 Interface of Customer data

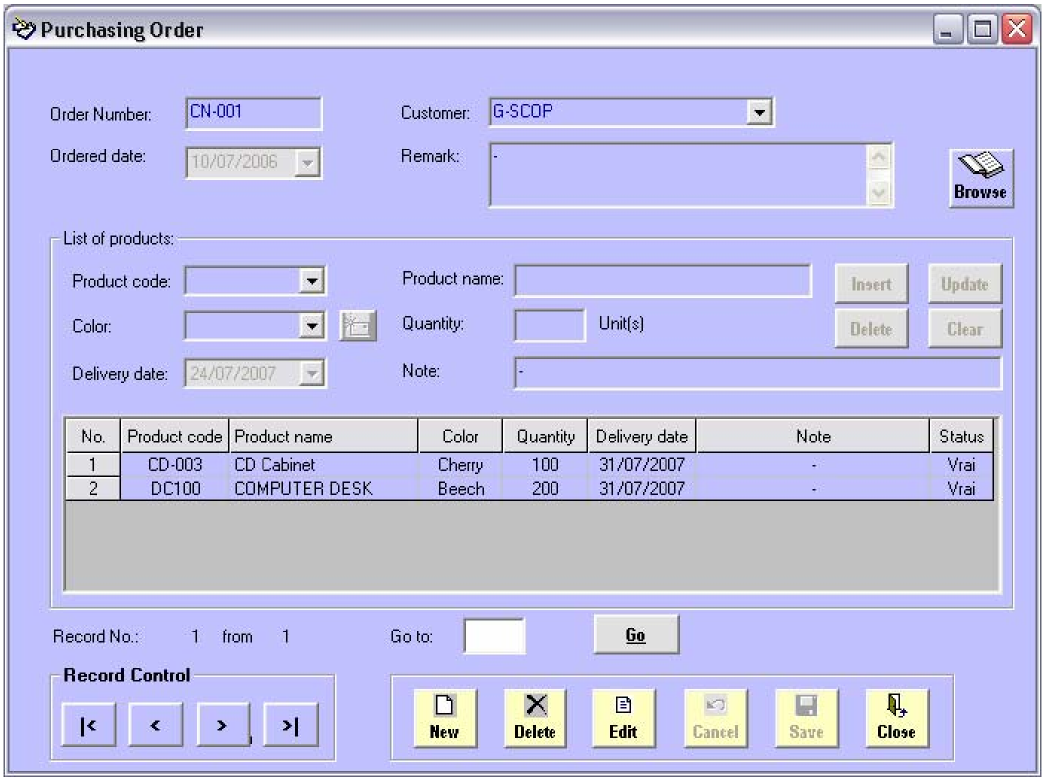


Figure 7.34 Interface of Purchasing Order

<u>Design</u>

This section is the initial part of problems, we have to solve. It contains three interfaces that store the information of *Product Data*, *Part Data*, and *Bill of Materials* (BOM). The interface of *Product Data* stores the product's specification, details of a product, list of parts and theirs dimensions as represented in Figure 7.35.

The interface of *Part Data* also presents specifications, and characteristics for each part, which mean the details of (raw) material that used to produce the part, as represented in Figure 7.36. This information is mainly contributed by the sketcher (the initial information of the conceptual design) or the person who concerns the aesthetic of the product, the thickness of parts contributed by the mechanic, and also the allowance value of defects (during the manufacturing process) in percentage that is normally contributed by the manufacturer.

The interface of *Bill of Materials* stores the list of materials (only purchased part, not raw materials) that needed to produce the product. The materials are mostly fasteners and packaging materials. Figure 7.37 represents the interface form of *Bill of Materials*. This information is mainly contributed by the assembler.

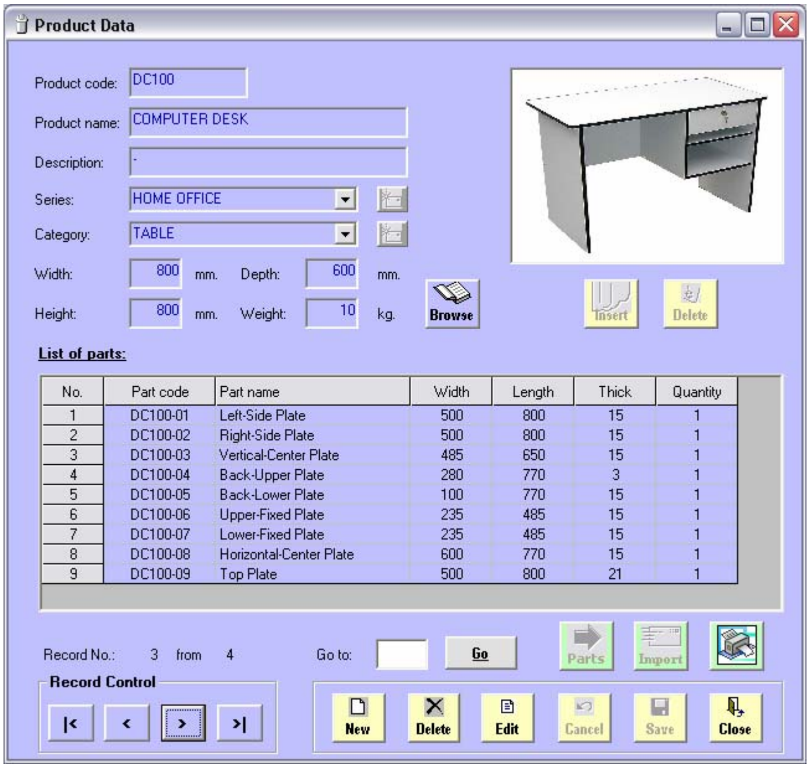


Figure 7.35 Interface of Product Data

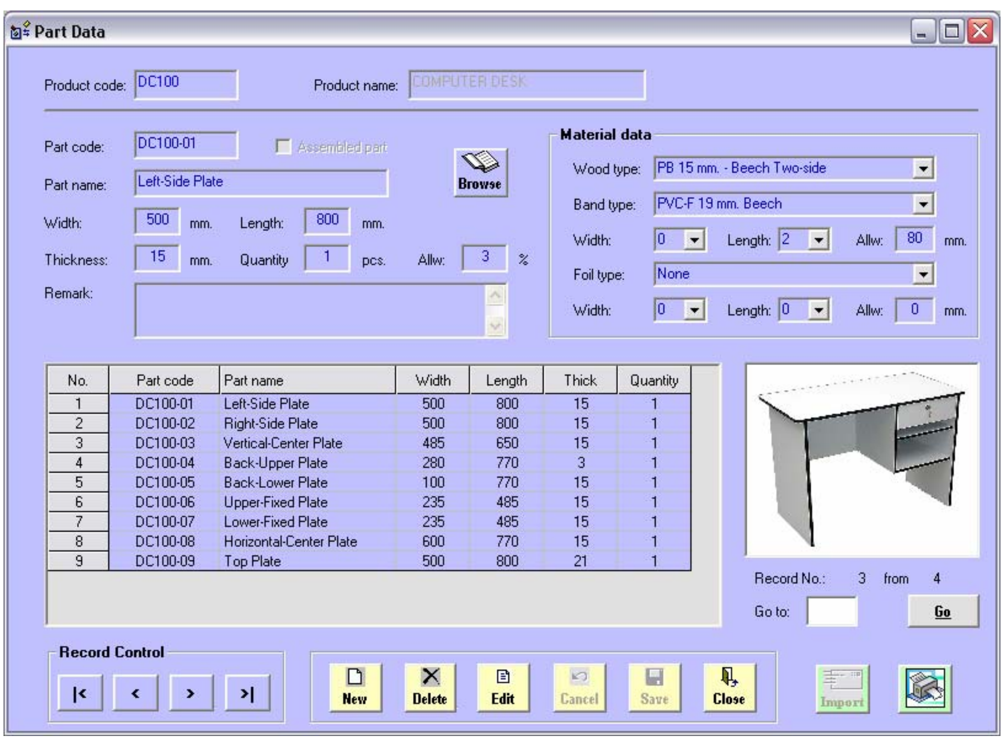


Figure 7.36 Interface of Part Data

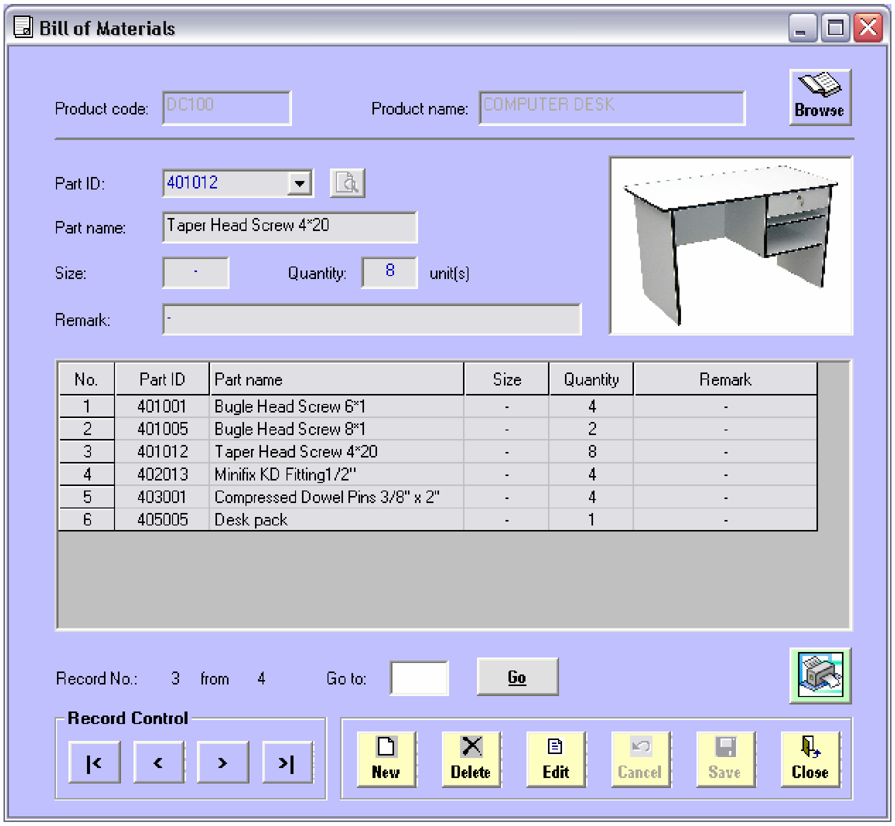


Figure 7.37 Interface of Bill of Materials

These three interfaces are corresponding to the *Planning* section. They support to create controlled documents such a *Production Order*, *Materials Requirement*, which will be presented in the section of *Output information*.

<u>Purchasing & Stock</u>

This section stores the information of raw materials and materials such purchased parts. The interface of raw materials includes particleboard and medium-density fiberboard (*PB-MDF*), *Band*, and *Foil* while the interface of materials is *Purchased Parts*. The interface of *PB-MDF* stores the list of available boards and theirs details, as represented in Figure 7.38.

The interface of *Band* and *Foil* stores the list and details of band and foil that used to cover the edges of a part as represented in Figure 7.39 and Figure 7.40 respectively. As same as the interface of raw materials, the interface of *Purchased Parts* stores the list of available purchased parts which are mostly fasteners and packaging materials as represented in Figure 7.41.

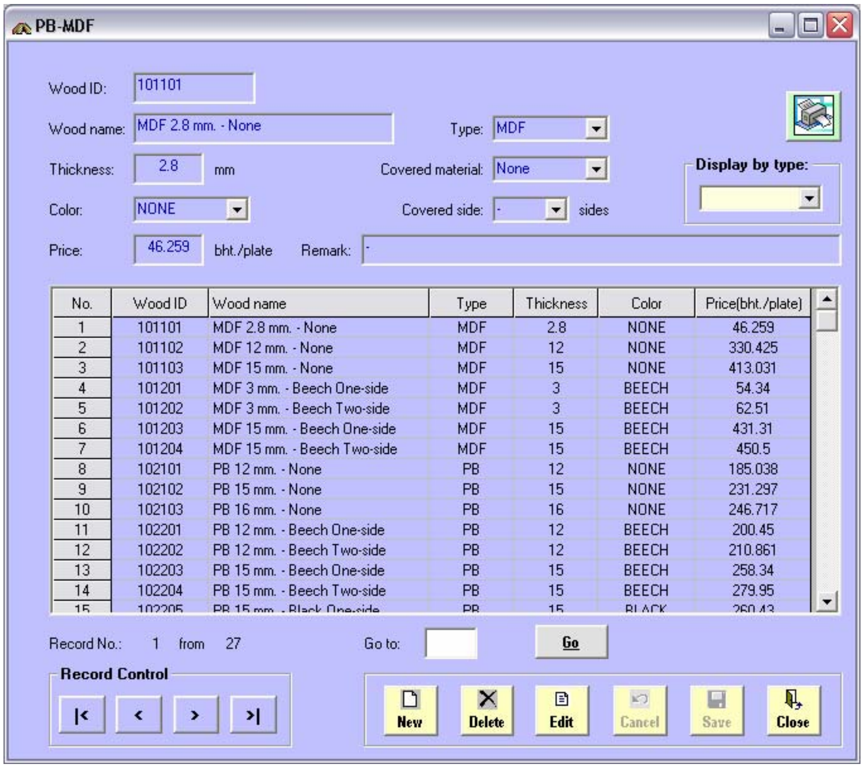


Figure 7.38 Interface of the list of particleboard and medium-density fiberboard

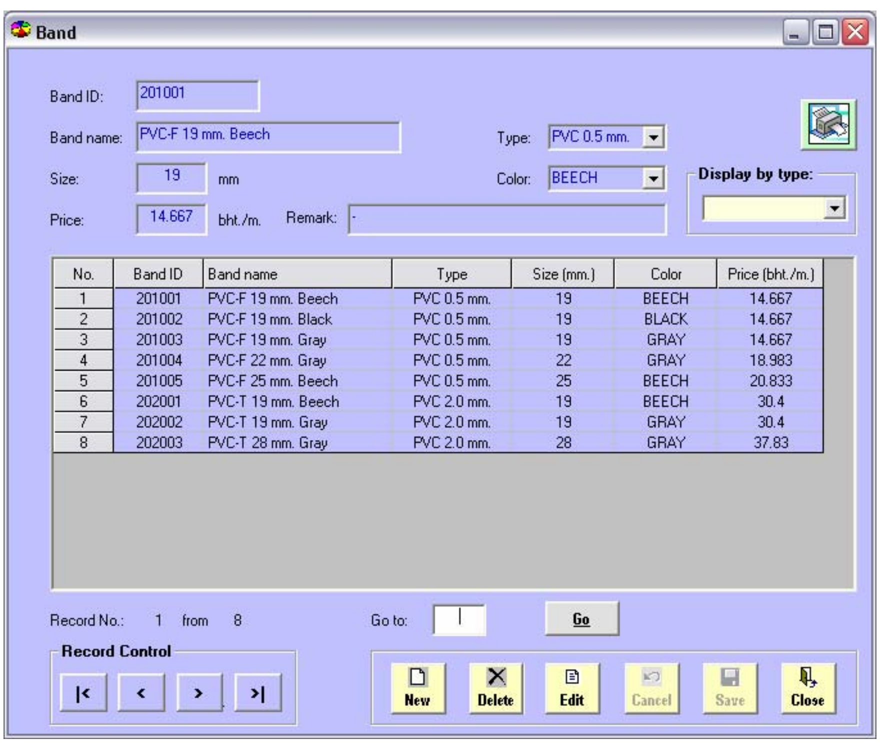


Figure 7.39 Interface of the list of band

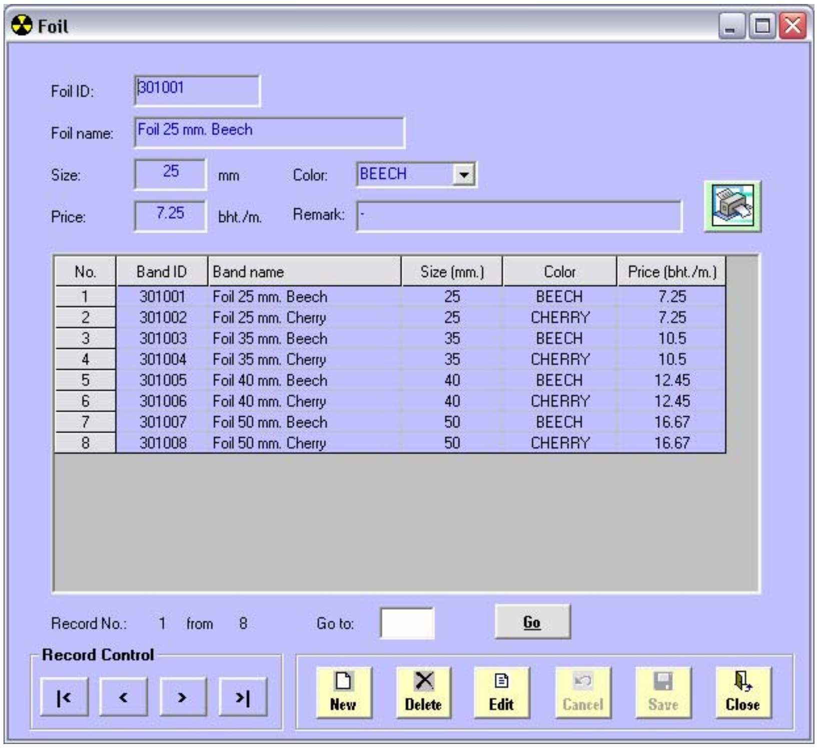


Figure 7.40 Interface of the list of foil

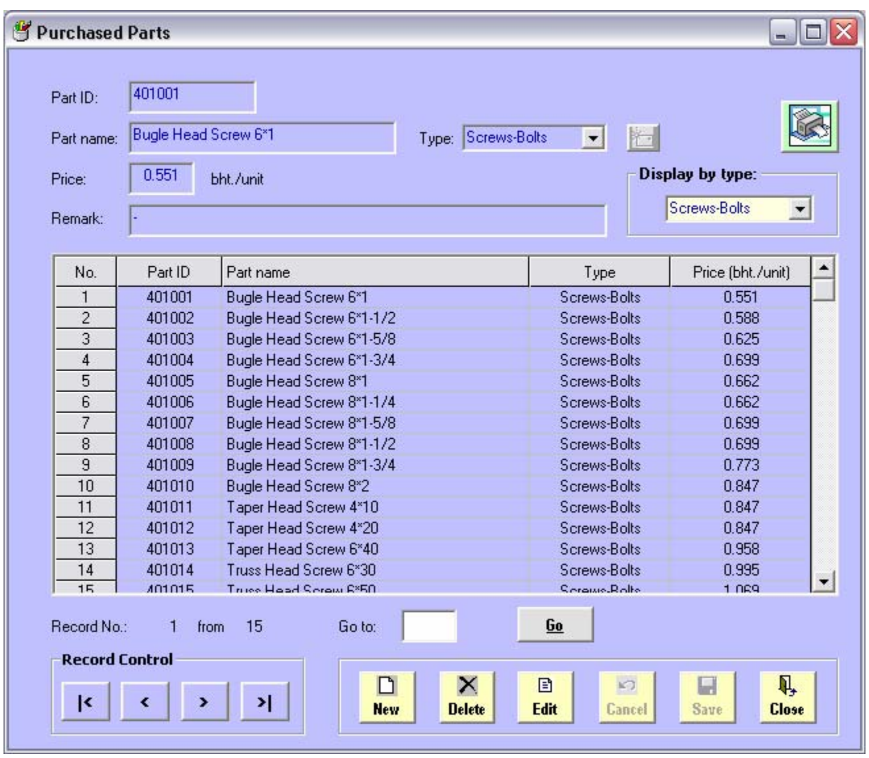


Figure 7.41 Interface of the list of purchased part

The information of this section supports the interfaces of *Part Data* and *Bill of Materials* in the Design section and facilitates the user to choose the available raw materials/ purchased parts.

<u>Engineering</u>

This section stores the information that enhances the manufacturer to contribute his/her knowledge. It contains six interfaces i.e. *Work Center*, *Machine*, *Standard Process*, *Process Route Library*, *Manufacturing Data*, and *Daily Report*. The interface of *Work Center* stores the information of groups of standard processes that are divided by the tasks and generally the layout of the plant. For example, the work center of *banding* contains two processes: *straight-banding* and *curve-banding*, that could be in the same working area. It also groups the machines that are in used to perform the same task and working area. The interface of *Work Center* is represented in Figure 7.42.

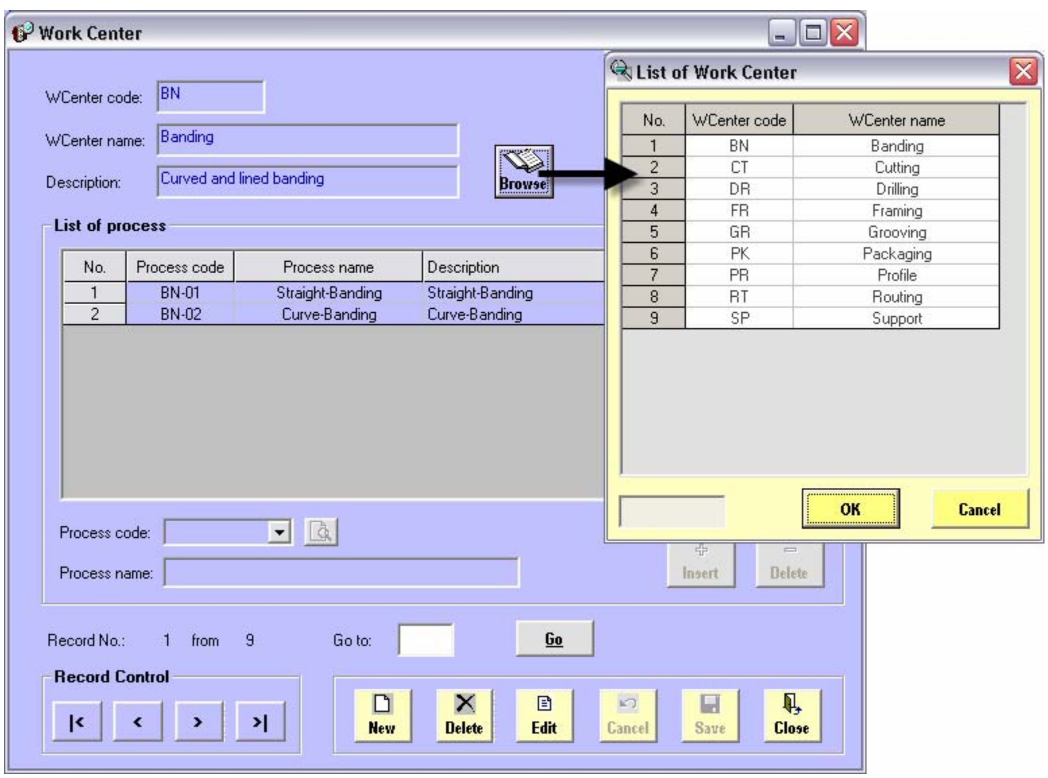


Figure 7.42 Interface of Work Center

The interface of *Machine* stores the list of the machines and indicates in which work center they are, as represented in Figure 7.43. The interface of *Standard Process* stores the list of manufacturing processes, as represented in Figure 7.44. These three interfaces contribute the data that supports the user to create process routes and to input the manufacturing data in the form *Daily Report*.

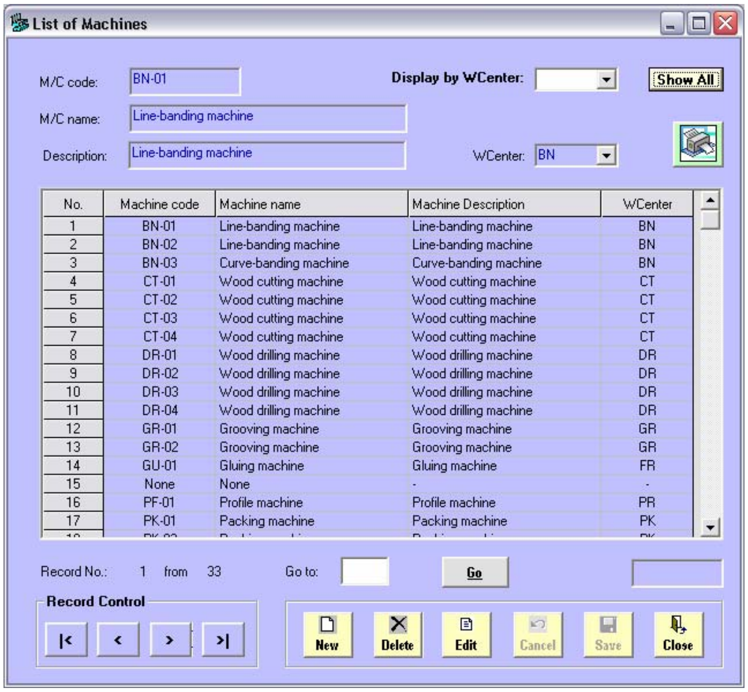


Figure 7.43 Interface of the list of machines

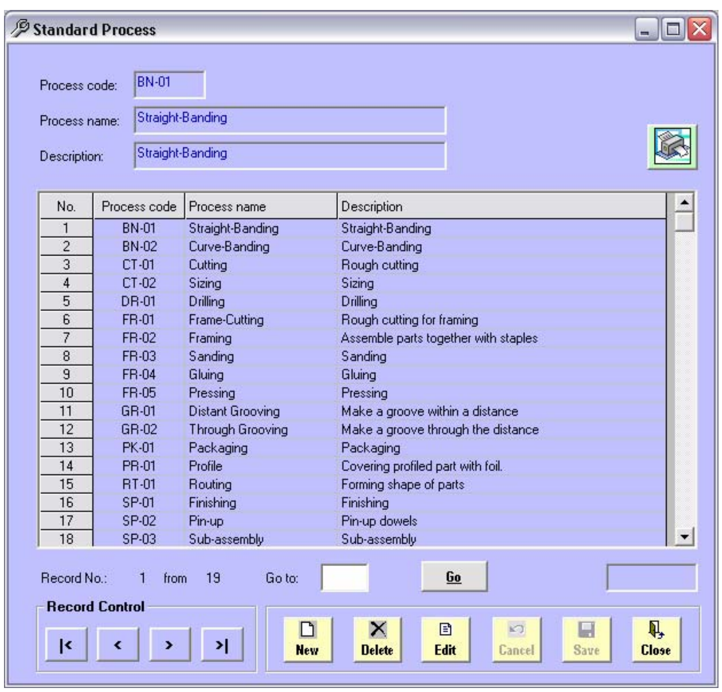


Figure 7.44 Interface of the list of standard processes

Process Route Library is a sort of knowledge base about manufacturing system. The interface of *Process Route Library* contains a list of process routes and each process route stores a list of manufacturing processes. This library is normally created by the manufacturer and is used to define for each part the processes which must be operated. It selects the most appropriate process route definition instead of define the process for each part one by one every time. In the case of a new model, there might not have an appropriate definition. The manufacturer must then define a new process route for those parts. Nevertheless, as much as s/he provides the definitions, the less time of define repeatly the manufacturing processes for parts. The interface of *Process Route Library* is represented in Figure 7.45.

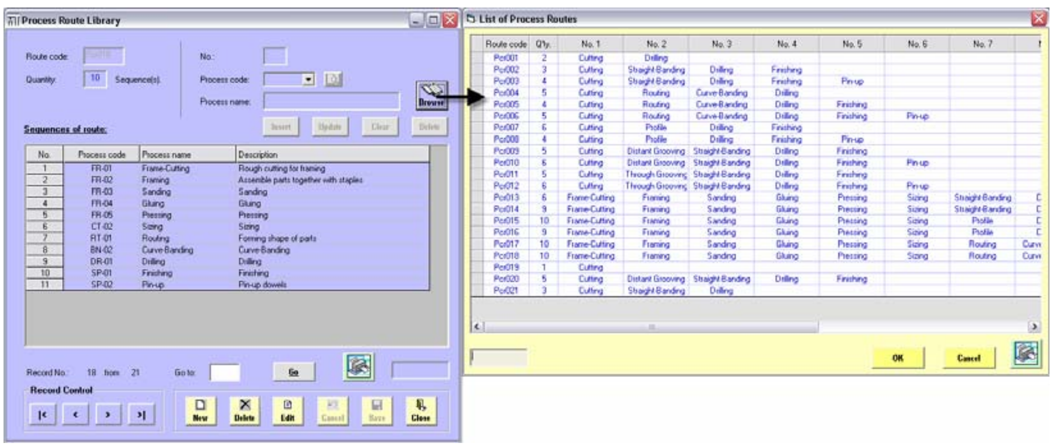


Figure 7.45 Interface of Process Route Library

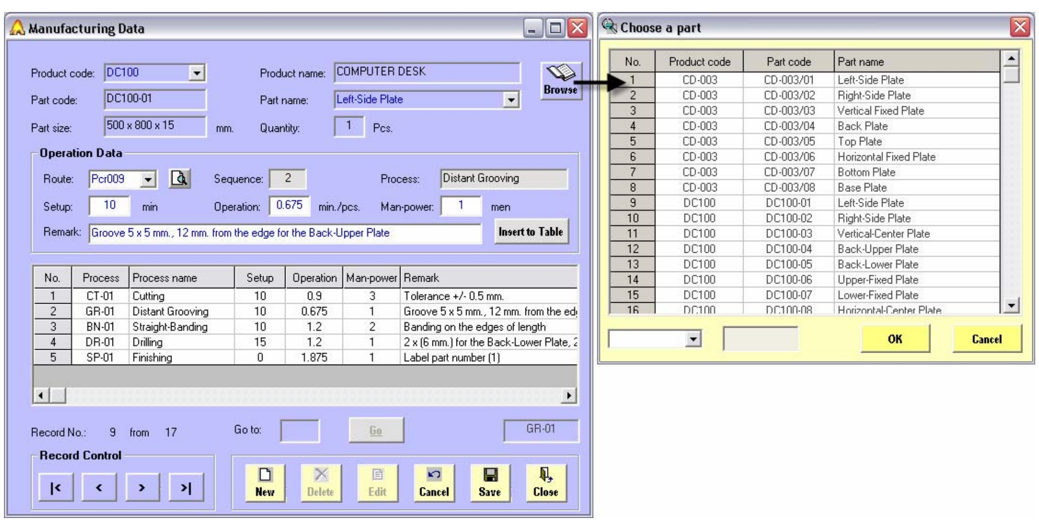


Figure 7.46 Interface of Manufacturing data

With the supporting information from *Product Data*, *Part Data*, and *Process Route Library*, the interface of *Manufacturing Data* permits the manufacturer to manipulate his/her knowledge about the product. This interface stores the manufacturing knowledge of a product and its parts. The manufacturer can define the *setup time*, *operation time*, *manpower*, and *remarks* for each process of each part, as represented in Figure 7.46. This information supports the manufacturer to evaluate the product i.e. time estimation and cost estimation, and to create the controlled documents in the section of *Planning*.

Note that to define the values such setup time, operation time, and manpower, the user must have experience or competence on that task. S/he may exploit the theory of motion and time study to acquire such values. [Feng and Song 2000] defines equations of manufacturing time estimating and setup activity time based on the Activity Based Costing (ABC) method as written in Equation (7.1) and (7.2) respectively.

Manufacturing time estimating:

$$t_m = \sum_{i=1}^{N} t_{activity}^{i} = \sum_{i=1}^{N} \left(t_{processing}^{i} + t_{setup}^{i} + t_{handling}^{i} + t_{load-unload}^{i} + t_{idling}^{i} \right) \quad (7.1)$$

Where: t_m - is the total estimated time of an artifact

$t_{activity}$ - is the estimated time of activity i

i - is an index

N - is the total number of manufacturing process

$t_{processing}^{i}$ - is the processing time of activity i

t_{setup}^{i} - is the setup time of activity i

$t_{handling}^{i}$ - is the handling time of activity i

$t_{load-unload}^{i}$ - is the load and unload time of activity i

t_{idling}^{i} - is idling time of activity i

Setup activity time:

$$t_{setup}^{i} = t_{s-machine}^{i} + t_{s-tool}^{i} + t_{s-workpiece}^{i} \quad (7.2)$$

Where: $t_{s-machine}^{i}$ - is the machine setup time of activity i

t_{s-tool}^{i} - is the tool setup time of activity i

$t_{s-workpiece}^{i}$ - is the work piece setup time of activity i

Nevertheless, DAPP estimates the time estimation based on the decomposition of the manufacturing process and process planning information. We may rewrite equations of time estimating using in DAPP as following:

$$t_m = \sum_{i=1}^{N} t_{activity}^i = \sum_{i=1}^{N}\left(t_{operation}^i + t_{setup}^i\right) \tag{7.3}$$

Where: $t_{operation}^i$ - is the operation time of activity i

The operation time estimating comprises of following activities:

$$t_{operation}^i = t_{s-workpiece}^i + t_{processing}^i + t_{handling}^i + t_{load-unload}^i + t_{idling}^i \tag{7.4}$$

The setup time is also changed as following:

$$t_{setup}^i = t_{s-machine}^i + t_{s-tool}^i \tag{7.5}$$

The equations used in DAPP are rather based on the practical manner that may not theoretically correct. We count the setup time of work piece into the operation time for the reason that in the practical way, the operators in manufacturing process can not record every single setup of work piece into the check sheet such *Daily Report* without trouble their operations, due to the short cycle time of the work piece. *Daily Report* is a sort of check sheet which is designed to support the manufacturer for collecting manufacturing data. To acquire such data, we develop an interface of *Daily Report* for collecting the real data during the manufacturing process. The interface of *Daily Report* permits the operators to collect the data of each work center, each machine by the support information of *Work Center*, *Machine*, and *Production Order*. Figure 7.47 represents the manufacturing data recorded by the operators in the production line, including the data of setup time, operation time, manpower, quantity of operated parts, remark, etc.

One of the objectives of DAPP is to monitor the manufacturing process and to analyze the results of manufacturing process. The results of analysis indicate the accuracy of the manufacturing plan and permit the manufacturer to improve his/her production planning following the real situation. Furthermore, such statistical information could be applied to define the allowance values of material utilization i.e. particleboard, MDF, edge-bands, which is used in MRP[23]. DAPP outputs the results of manufacturing records as *Production Reports* that will be presented in the part of *Output information*.

[23] Material Requirement Planning

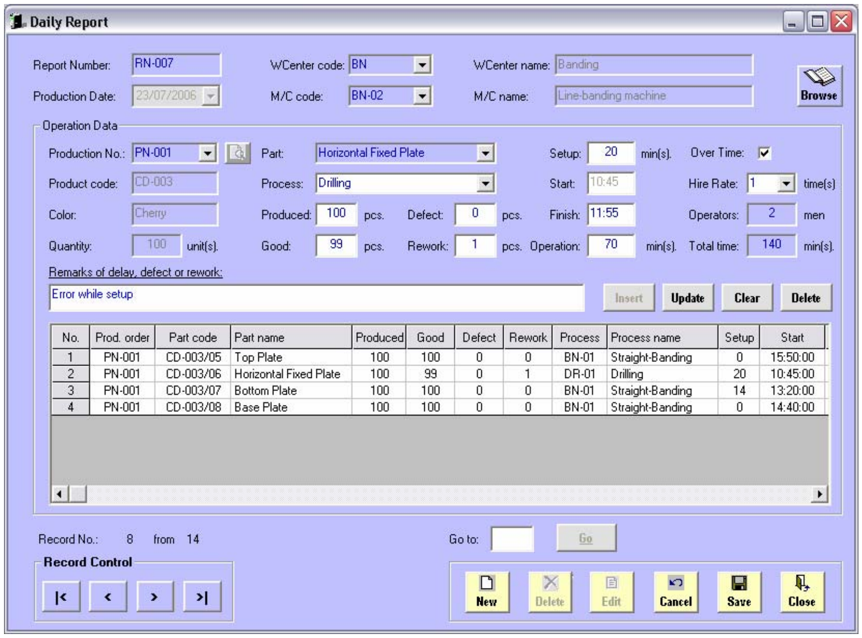


Figure 7.47 Interface of Daily Report

<u>Human Resource</u>

This section stores the list of employees. The interface of *Employee Data* stores the information of workers who operate in the manufacturing processes as represented in Figure 7.48. It concerns mainly what process they are working, the corresponding work center, and their salaries (per day) they earn to define the labor cost of the production.

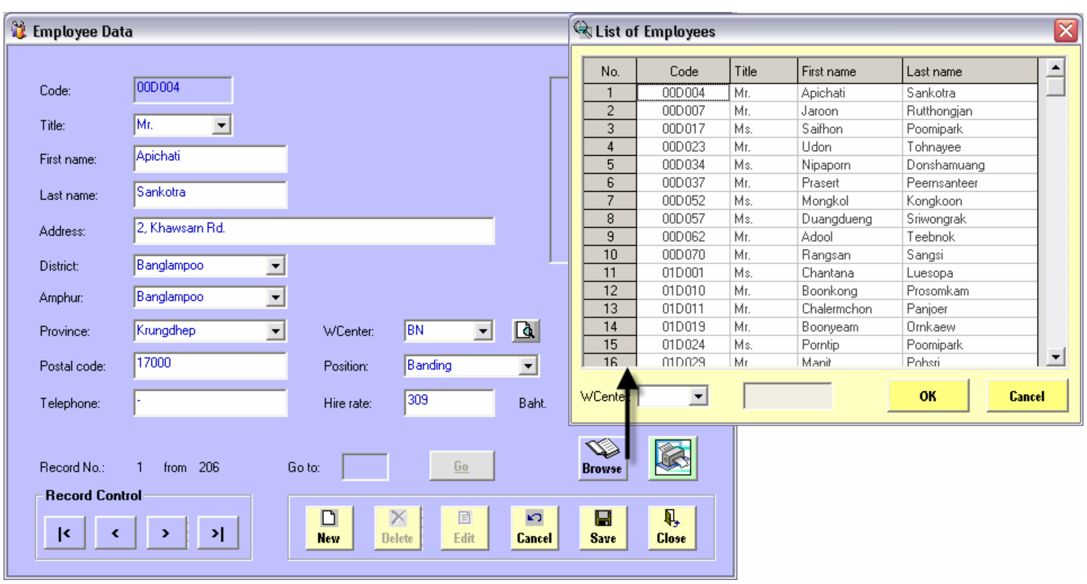


Figure 7.48 Interface of Employee data

7.5.3 Output information

The *Output information* comprises only the *Planning* section. This section satisfies the objectives of DAPP that are: to contribute the results of the evaluation and to contribute the information for developing the process planning. This section includes interfaces of *Production Order*, *Job Order*, *Material Requirements*, *Time Estimation*, *Cost Estimation*, and *Production Report*.

The interfaces of *Production Order*, *Job Order*, *Material Requirements*, *Time Estimation*, and *Cost Estimation* satisfy both of the two principal objectives of DAPP but it contributes mainly the information that can be used to evaluate the design and to plan the production before performing it. The interfaces of *Production Report* concern rather satisfying the industrial part. It analyzes the records of manufacturing process that are contributed by the *Daily Report*.

<u>Production Order and Job Order</u>

To create a conceptual process planning, we use the interface of *Production Order* and *Job Order* to represent the results of the input information. In fact, the *Production Order* satisfies rather the industrial part. The manufacturer uses the *Production Order* as a controlled document in the production process. When the user chooses the purchasing order and the item that s/he wants to produce by the interface of *Production Order*, it will represent automatically the details of the item, starting date, finished date, delivery date and also a list of parts, its dimension and quantity, as represented in Figure 7.49.

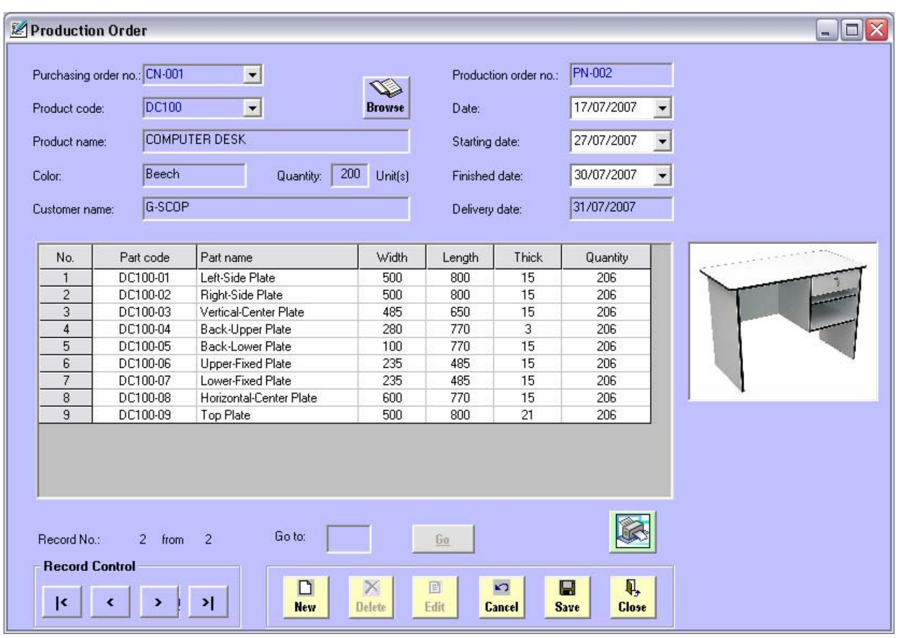


Figure 7.49 Interface of Production Order

The *Production Order* gives general details of the item while the *Job Order* represents the details of the parts of the selected item. The *Job Order* satisfies both of the industrial part and the design process. It creates a conceptual process plan by listing the processes and the *work centers* where they must be operated, and also the details of operations as represented in Figure 7.50. It is used as a controlled document in the manufacturing process. To facilitate the manufacturer, DAPP creates automatically printable controlled documents of *Production Order* and *Job Order* that can be found in Annex II.

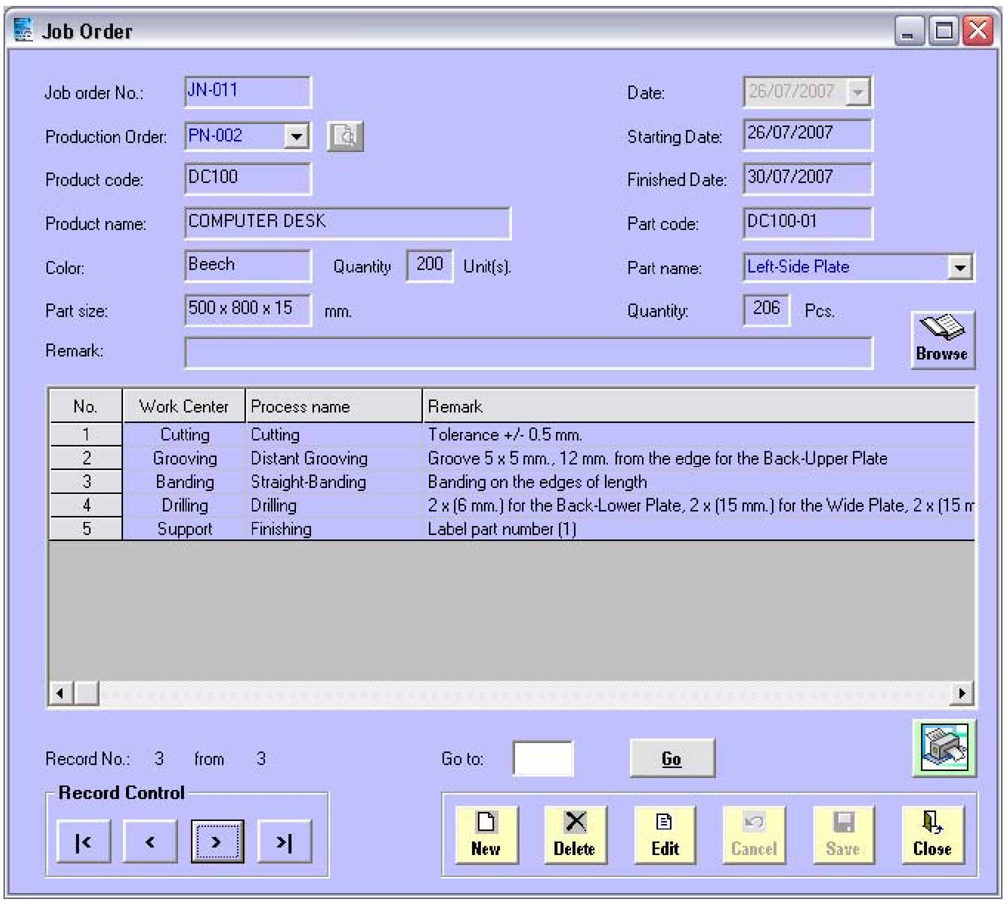


Figure 7.50 Interface of Job Order

Materials Requirement

The *Materials Requirement* is as important as the process planning. It calculates the required materials that must be used for producing a product. This information enhances the production planner to plan the Materials Requirement Planning (MRP). The interface of *Materials Requirement* represents both of a list of raw materials (PB/MDF, band, and/or foil) and a list of purchased parts of the selected product, as represented in Figure 7.51.

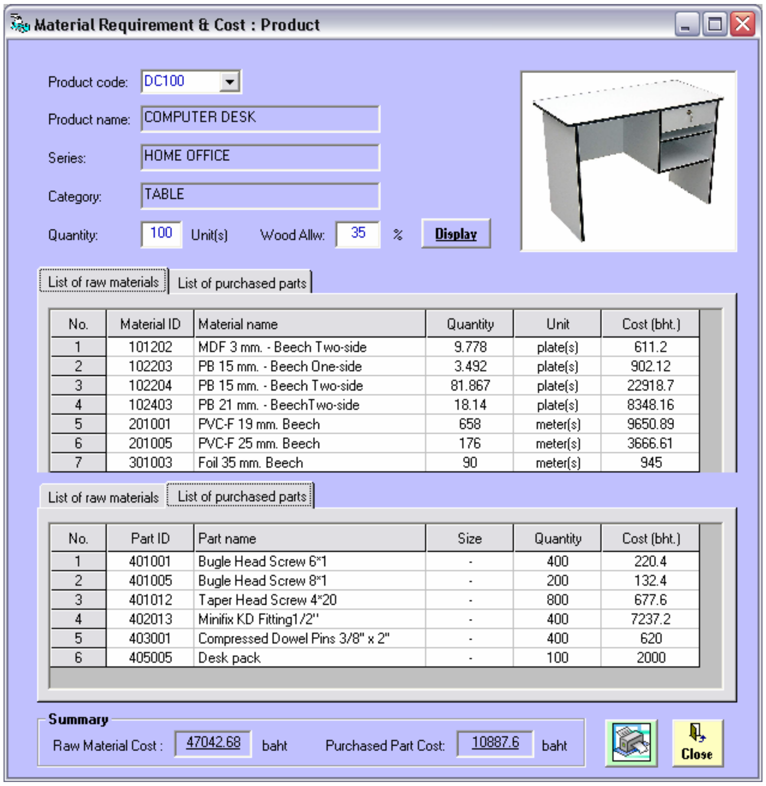


Figure 7.51 Interface of materials requirement represented by product

Furthermore, it also estimates the cost of the required materials. We can estimate the material cost of a product per unit by formulating these following equations:

$$C_{MAT} = C_{RM} + C_{PP} \tag{7.6}$$

Where: C_{MAT} - is the cost of materials requirement

C_{RM} - is the cost of raw materials

C_{PP} - is the cost of purchased parts

The cost of raw materials can be formulated by these equations:

$$C_{RM} = N \times \left(C_B + C_E + C_F\right) \tag{7.7}$$

Where: N - is quantity of the work pieces of the part

C_B - is the cost of PB/MDF

C_E - is the cost of band

C_F - is the cost of foil

$$C_B = \left(\frac{p_w \times p_l}{B_W \times B_L}\right) \times \left(\frac{100 + B_{AW}}{100}\right) \times B_P \tag{7.8}$$

$$C_E = \left[\frac{n_w\left(p_w + E_{AWW}\right) + n_l\left(p_l + E_{AWL}\right)}{1000} \right] \times E_P \qquad (7.9)$$

$$C_F = \left[\frac{n_w\left(p_w + F_{AWW}\right) + n_l\left(p_l + F_{AWL}\right)}{1000} \right] \times F_P \qquad (7.10)$$

Where: p_w - is the width of the work pieces

p_l - is the length of the work pieces

B_W - is the width of the board

B_L - is the length of the board

B_{AW} - is the allowance value of board utilization [24]

B_P - is the price of the board (price per unit)

n_w - is the number of covered wide-sides

n_l - is the number of covered long-sides

E_{AWW} - is the allowance value of band utilization (wide-side)

E_{AWL} - is the allowance value of band utilization (long-side)

E_P - is the price of the band (price per meter)

F_{AWW} - is the allowance value of foil utilization (wide-side)

F_{AWL} - is the allowance value of foil utilization (long-side)

F_P - is the price of the band (price per meter)

The cost of purchased parts can be formulated by a simple equation:

$$C_{PP} = N_{PP} \times P_{pp} \qquad (7.11)$$

Where: N_{PP} - is quantity of the purchased part using per one unit

P_{PP} - is the price of the purchased part (price per part)

As a result, the cost estimation of materials requirement is then calculated. In addition, to satisfy the industrial part, the *Materials Requirement* also represents the list of materials requirement following the purchasing order. The user can choose the purchasing order to represent the list of materials requirement of all items in the purchasing order as represented in Figure 7.52, or represent only the selected item as represented in Figure 7.53. To facilitate the production planner, DAPP also creates printable controlled documents of materials requirement as represented in Annex II.

[24] This value is calculated by the manufacturer or the person who is in charge of cutting layout planning. The value varies depending on the cutting layout planning.

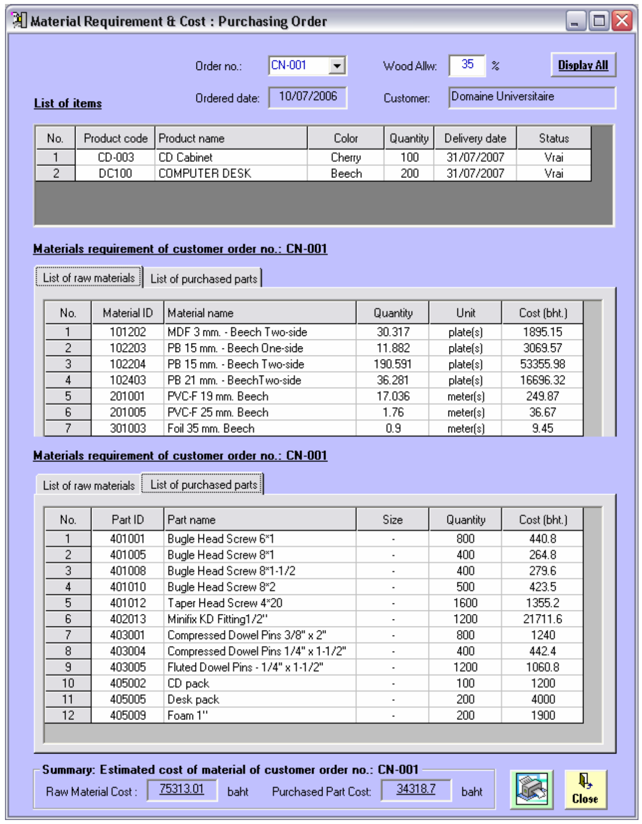


Figure 7.52 Interface of materials requirement represented by purchasing order

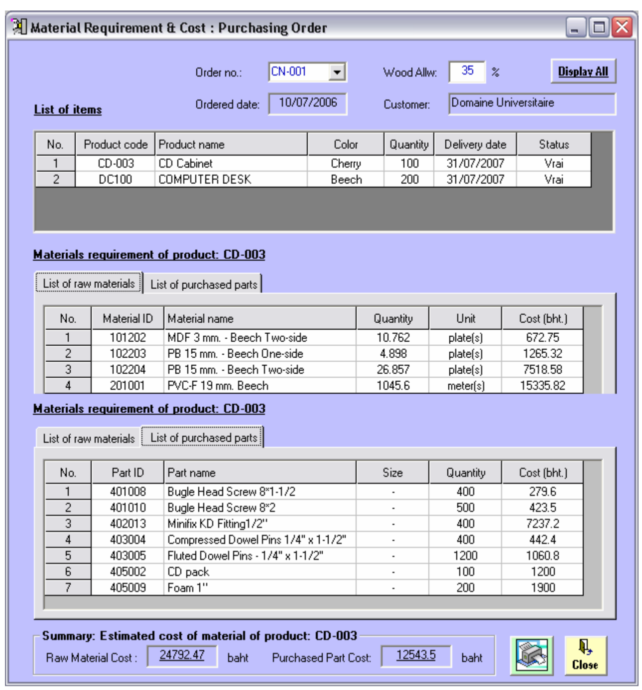


Figure 7.53 Interface of materials requirement represented by selected item

<u>Time Estimation</u>

To organize the production planning, one of the most important information is that the time estimation of manufacturing. According to the contributed information from the interface of *Manufacturing Data* as presented in section 7.5.2, we can estimate the manufacturing time (setup time and operation time) of a product. DAPP provides the user the interface of *Time Estimation* that represents the estimation of setup time and operation time of a product with given quantity as represented in Figure 7.54. It also represents the summary of time estimation represented by parts and by process as represented in Figure 7.55.

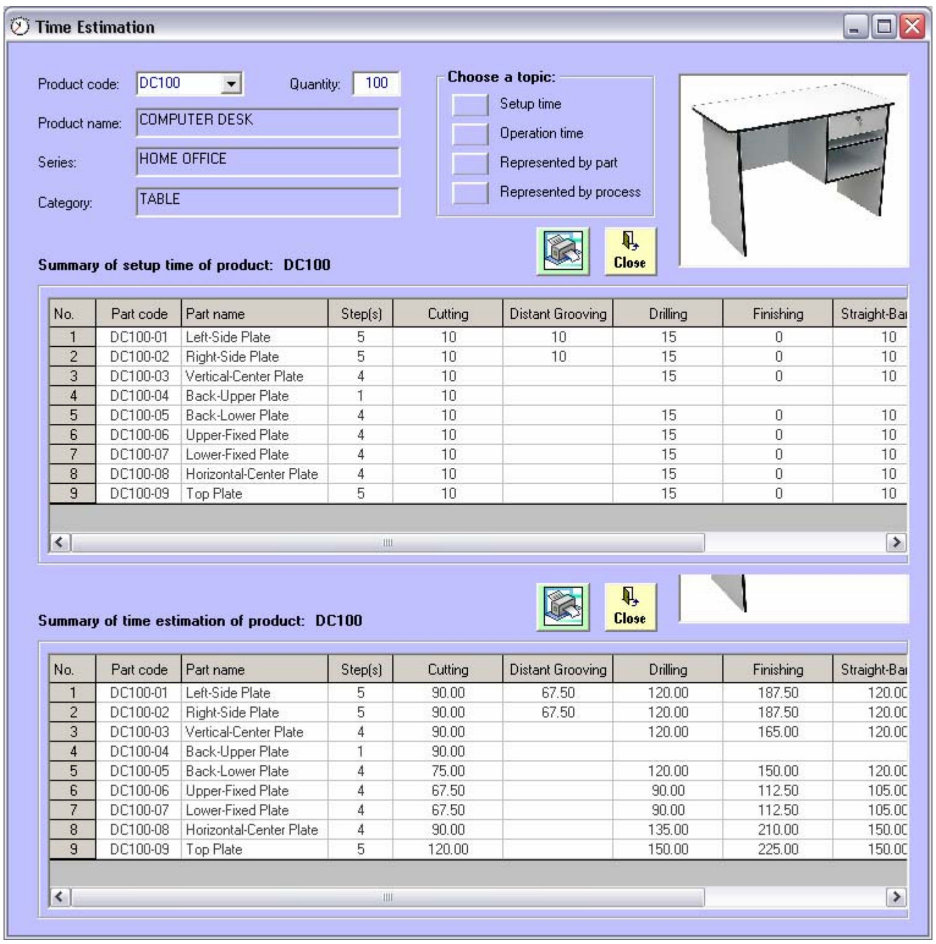


Figure 7.54 Interface of the estimation of setup time and operation time

As well as the interface of *Materials Requirement*, this interface facilitates the production planner by providing the printable controlled documents of time estimation as represented in Annex II. In the mean time, DAPP creates automatically the chart representing the summary of time estimation represented by part and by process as bar charts as represented in Annex II. These bar charts give the manufacturer a visual differential time consuming between the parts and the processes. This information enhances the manufacturer to develop the manufacturing process.

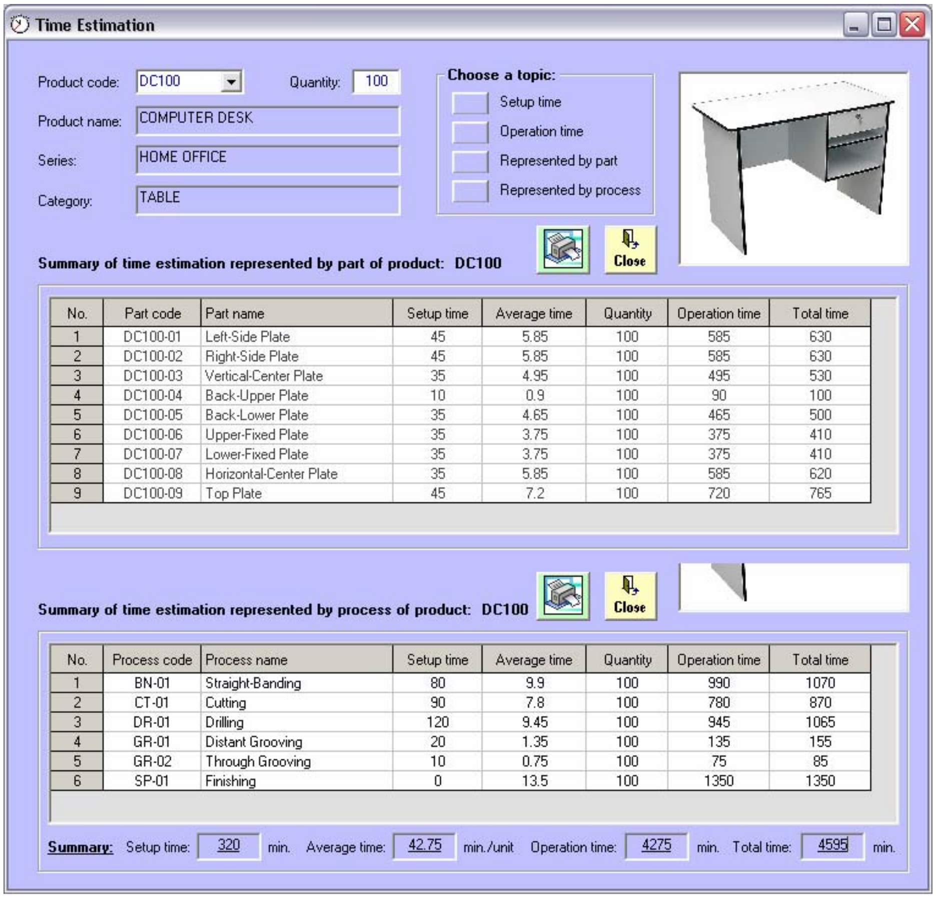


Figure 7.55 Interface of the estimation time represented by parts and process

Cost Estimation

The cost is the most critical factor that influences the design team in the ultimate decision. It is obliged to assess the product cost as early as possible. DAPP estimates the cost of product base on a decomposition of the manufacturing process and process planning information, which called *generative cost estimation* method [Lutters et al 2006]. We can determine the cost of product by the following equation:

$$C_P = C_{MFG} + C_{MAT} \qquad (7.12)$$

Where: C_P - is the cost of product

C_{MFG} - is the cost of manufacturing

C_{MAT} - is the cost of materials requirement

Note that this product cost has not taken into account yet the administrative cost, the transportation cost, and the other cost that are not involved in the production. [Feng and Song 2000] defines manufacturing cost estimating equations based on the Activity Based Costing (ABC) method as following:

$$C_{MFG} = \sum_{i=1}^{N} C_{activity}^{i}$$
$$= \sum_{i=1}^{N} \left(C_{processing}^{i} + C_{setup}^{i} + C_{handling}^{i} + C_{load-unload}^{i} + C_{idling}^{i} + C_{overhead}^{i} \right) \qquad (7.13)$$

Where the meaning of variables in Equation (7.13) is as same as in Equation (7.1) but it is replaced by '*cost*' instead of '*time*'. Theoretically, we have to include the overhead cost to this cost estimating. However, in the practical way, the overhead cost comprises the depreciation cost, the cost of maintenance of machines, the cost of public utility cost such electricity, water, etc., or may include some renting costs. Such costs are rather managed by the administrative section such accounting. In DAPP, we estimate the manufacturing cost based on the check sheet, *Daily Report*. As presented in the *manufacturing time estimating*, we can consider the cost of manufacturing as following:

$$C_{MFG} = \sum_{i=1}^{N} C_{activity}^{i} = \sum_{i=1}^{N} \left(C_{operation}^{i} + C_{setup}^{i} \right) \qquad (7.14)$$

Where the meaning of variables in Equation (7.14) is as same as in Equation (7.3) but it is replaced by '*cost*' instead of '*time*'. DAPP results merely the estimated cost of product based on the manufacturing and materials requirement as represented in Figure 7.56.

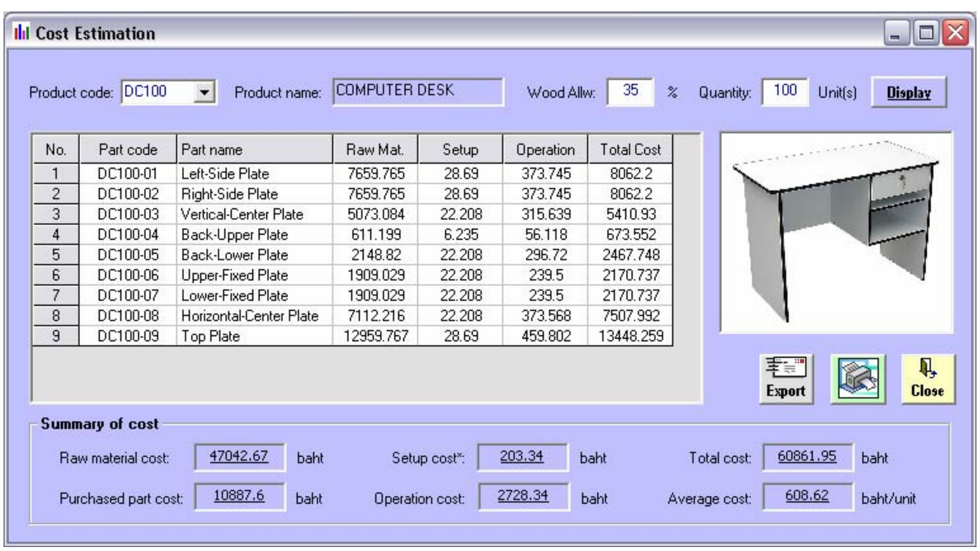


Figure 7.56 Interface of Cost Estimation

As well as the time estimation, DAPP offers the user to print out the summary of the cost estimation representing as bar charts as represented in Annex II. These bar charts give the manufacturer a visual differential cost of the parts. This information enhances manufacturer to develop the manufacturing process and also enhances the design team to develop the product design.

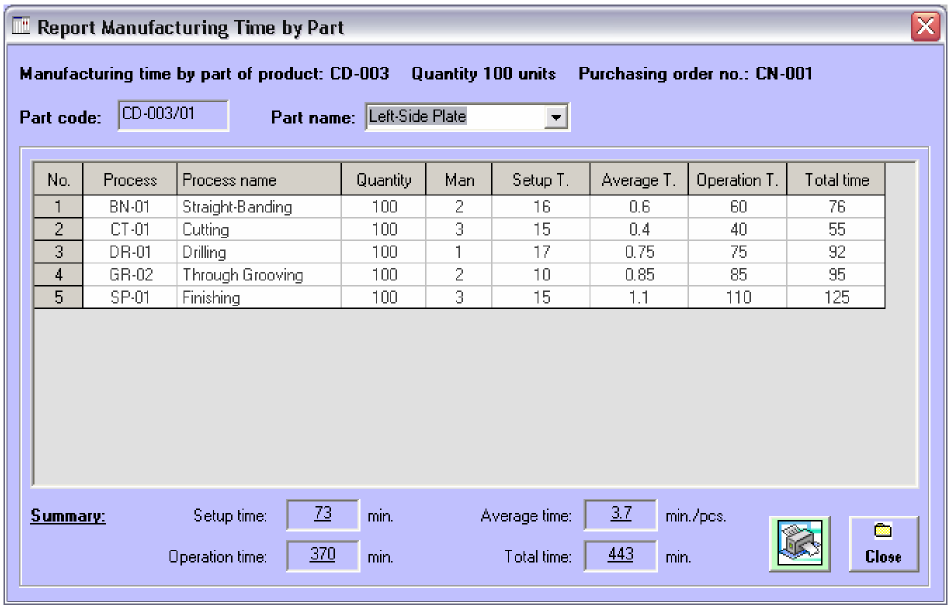
Production Report

This section concerns to satisfy the industrial part. It analyzes the records of manufacturing process that contributed by the *Daily Reports*. The objective of this section is to enhance the production planner to develop the process planning. It contains four groups of reports i.e. *reports of manufacturing time*, *reports of manufacturing cost*, *additional reports*, *reports of defects and reworks*, as represented in Figure 7.57. The user chooses the purchasing order and then selects the desired item that s/he wants to find out the reports.

Figure 7.57 Main interface of Production Report

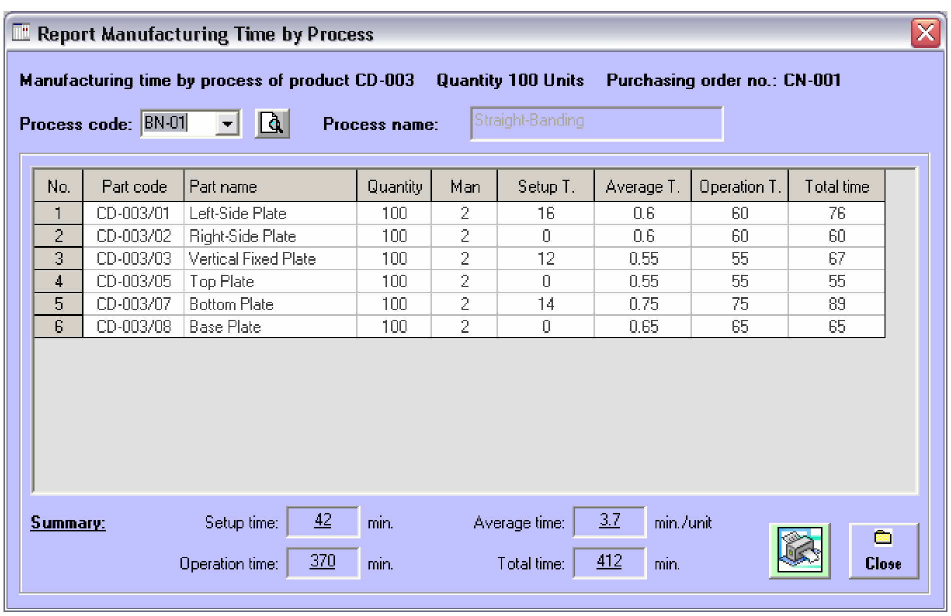
Reports of manufacturing time

This group reports the manufacturing time that includes setup time, operation time, total manufacturing time, and average manufacturing time. It can represent following the parts as represented in Figure 7.58, following the process as represented in Figure 7.59. To view all over the item, it represents the summary manufacturing time following the parts as represented in Figure 7.60, and following the process as represented in Figure 7.61. The user can print out these reports as same as the interface of time and cost estimation. The examples of printouts of these reports are represented in Annex II. As well as the *Time Estimation*, DAPP represents the reports in the printouts as bar charts that give the manufacturer a visual differential time consuming between the part and the process.



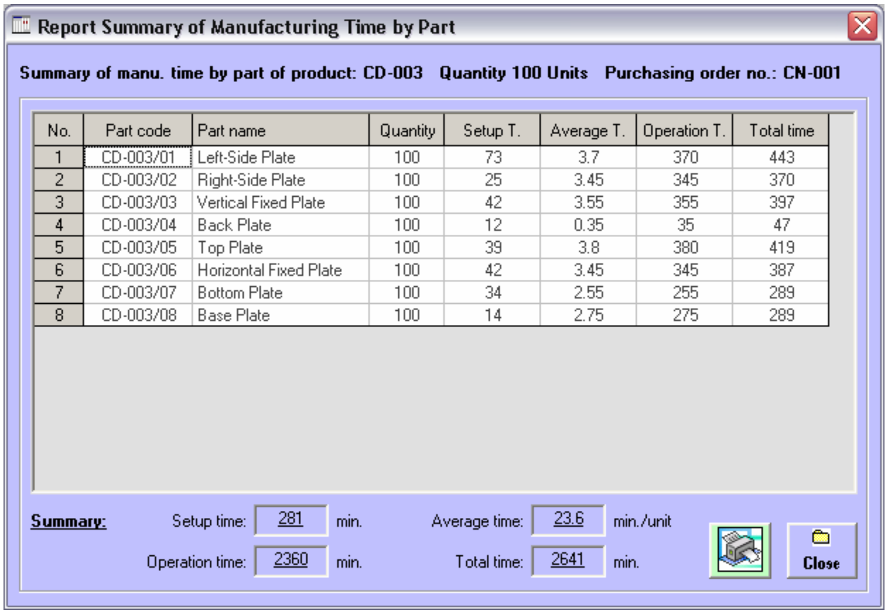
No.	Process	Process name	Quantity	Man	Setup T.	Average T.	Operation T.	Total time
1	BN-01	Straight-Banding	100	2	16	0.6	60	76
2	CT-01	Cutting	100	3	15	0.4	40	55
3	DR-01	Drilling	100	1	17	0.75	75	92
4	GR-02	Through Grooving	100	2	10	0.85	85	95
5	SP-01	Finishing	100	3	15	1.1	110	125

Figure 7.58 Report of manufacturing time represented by parts



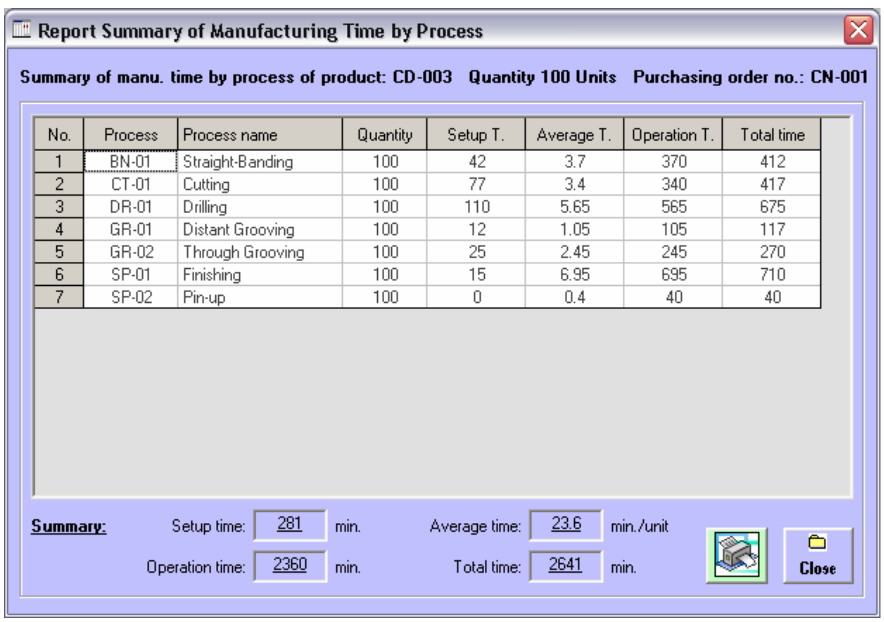
No.	Part code	Part name	Quantity	Man	Setup T.	Average T.	Operation T.	Total time
1	CD-003/01	Left-Side Plate	100	2	16	0.6	60	76
2	CD-003/02	Right-Side Plate	100	2	0	0.6	60	60
3	CD-003/03	Vertical Fixed Plate	100	2	12	0.55	55	67
4	CD-003/05	Top Plate	100	2	0	0.55	55	55
5	CD-003/07	Bottom Plate	100	2	14	0.75	75	89
6	CD-003/08	Base Plate	100	2	0	0.65	65	65

Figure 7.59 Report of manufacturing time represented by process



No.	Part code	Part name	Quantity	Setup T.	Average T.	Operation T.	Total time
1	CD-003/01	Left-Side Plate	100	73	3.7	370	443
2	CD-003/02	Right-Side Plate	100	25	3.45	345	370
3	CD-003/03	Vertical Fixed Plate	100	42	3.55	355	397
4	CD-003/04	Back Plate	100	12	0.35	35	47
5	CD-003/05	Top Plate	100	39	3.8	380	419
6	CD-003/06	Horizontal Fixed Plate	100	42	3.45	345	387
7	CD-003/07	Bottom Plate	100	34	2.55	255	289
8	CD-003/08	Base Plate	100	14	2.75	275	289

Figure 7.60 Summary report of manufacturing time represented by part

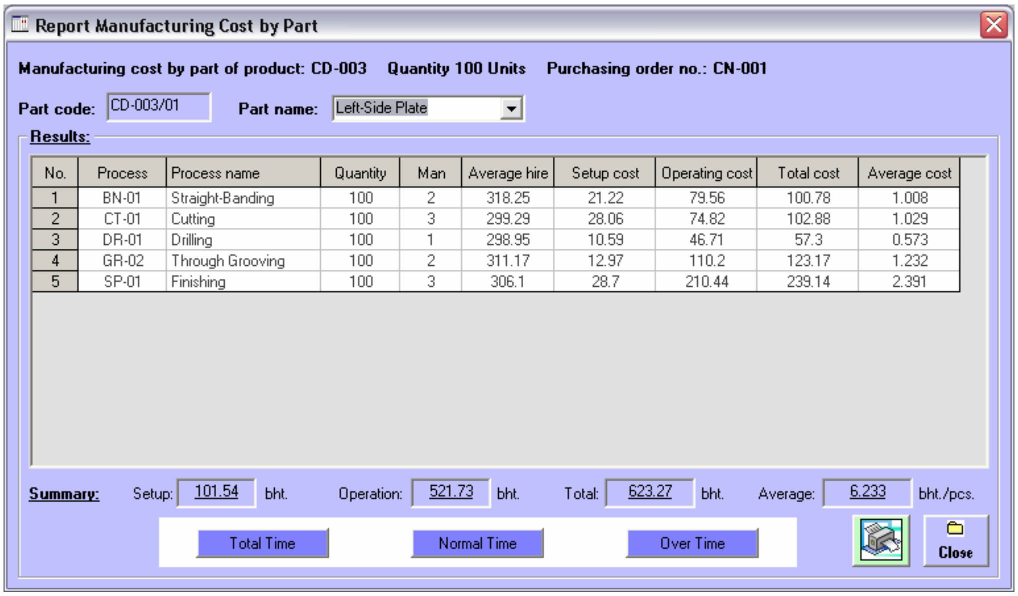


No.	Process	Process name	Quantity	Setup T.	Average T.	Operation T.	Total time
1	BN-01	Straight-Banding	100	42	3.7	370	412
2	CT-01	Cutting	100	77	3.4	340	417
3	DR-01	Drilling	100	110	5.65	565	675
4	GR-01	Distant Grooving	100	12	1.05	105	117
5	GR-02	Through Grooving	100	25	2.45	245	270
6	SP-01	Finishing	100	15	6.95	695	710
7	SP-02	Pin-up	100	0	0.4	40	40

Figure 7.61 Summary report of manufacturing time represented by process

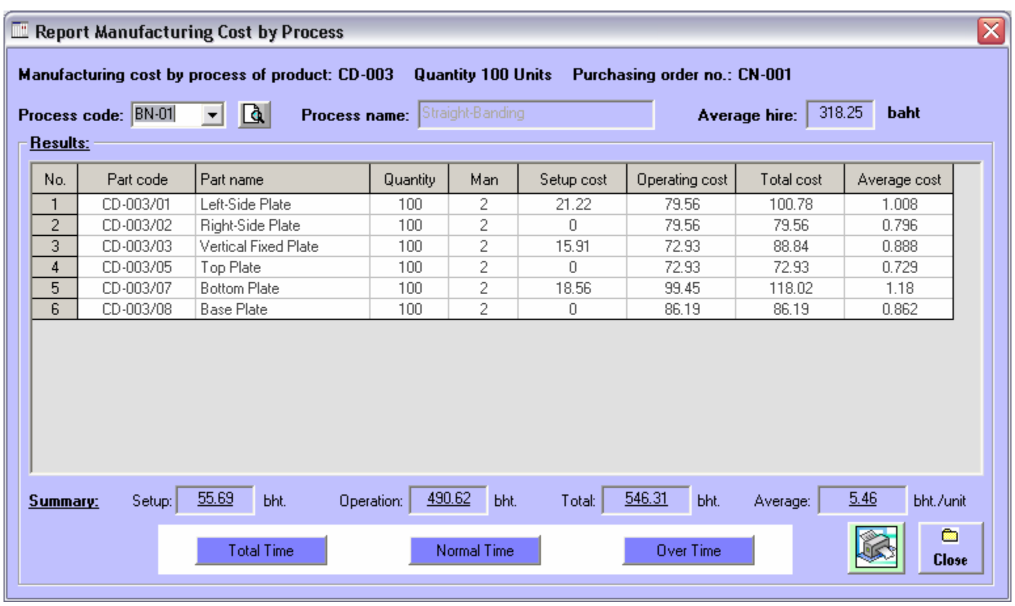
Reports of manufacturing cost

This group reports the manufacturing cost that includes setup cost, operation cost, total manufacturing cost, and average manufacturing cost per piece or unit. It can represent following the parts as represented in Figure 7.62, following the process as represented in Figure 7.63. To view all over the item, it represents the summary manufacturing cost following the parts as represented in Figure 7.64, and following the process as represented in Figure 7.65. As well as the reports of manufacturing time, the user can print out these reports. The examples of printouts of these reports are represented in Annex II. These reports are also represented as bar charts. This information enhances the manufacturer to determine the priority of which process that has to be improved.



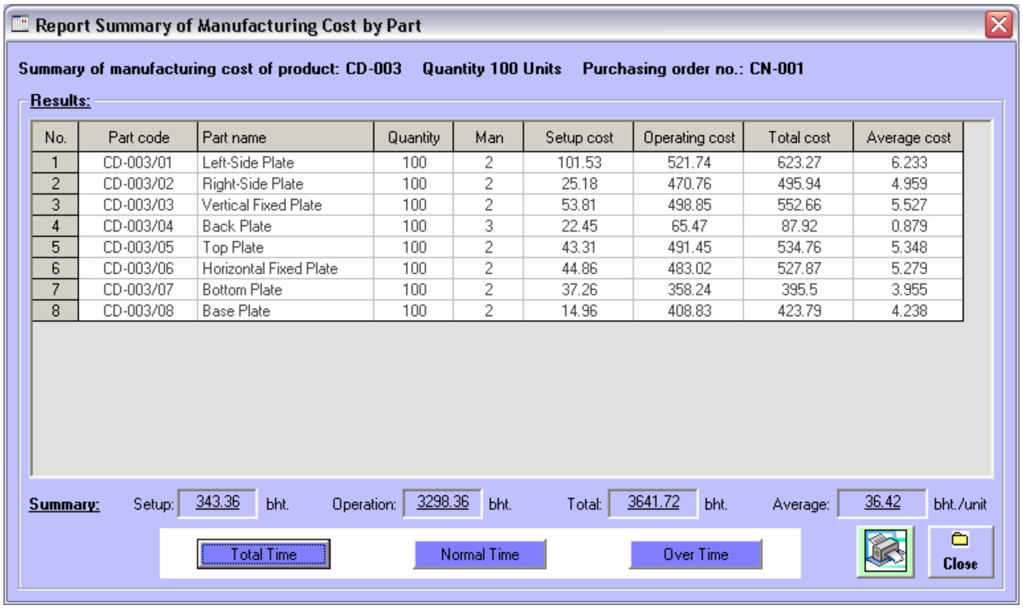
No.	Process	Process name	Quantity	Man	Average hire	Setup cost	Operating cost	Total cost	Average cost
1	BN-01	Straight-Banding	100	2	318.25	21.22	79.56	100.78	1.008
2	CT-01	Cutting	100	3	299.29	28.06	74.82	102.88	1.029
3	DR-01	Drilling	100	1	298.95	10.59	46.71	57.3	0.573
4	GR-02	Through Grooving	100	2	311.17	12.97	110.2	123.17	1.232
5	SP-01	Finishing	100	3	306.1	28.7	210.44	239.14	2.391

Figure 7.62 Report of manufacturing cost represented by part



No.	Part code	Part name	Quantity	Man	Setup cost	Operating cost	Total cost	Average cost
1	CD-003/01	Left-Side Plate	100	2	21.22	79.56	100.78	1.008
2	CD-003/02	Right-Side Plate	100	2	0	79.56	79.56	0.796
3	CD-003/03	Vertical Fixed Plate	100	2	15.91	72.93	88.84	0.888
4	CD-003/05	Top Plate	100	2	0	72.93	72.93	0.729
5	CD-003/07	Bottom Plate	100	2	18.56	99.45	118.02	1.18
6	CD-003/08	Base Plate	100	2	0	86.19	86.19	0.862

Figure 7.63 Report of manufacturing cost represented by process



Report Summary of Manufacturing Cost by Part

Summary of manufacturing cost of product: CD-003 Quantity 100 Units Purchasing order no.: CN-001

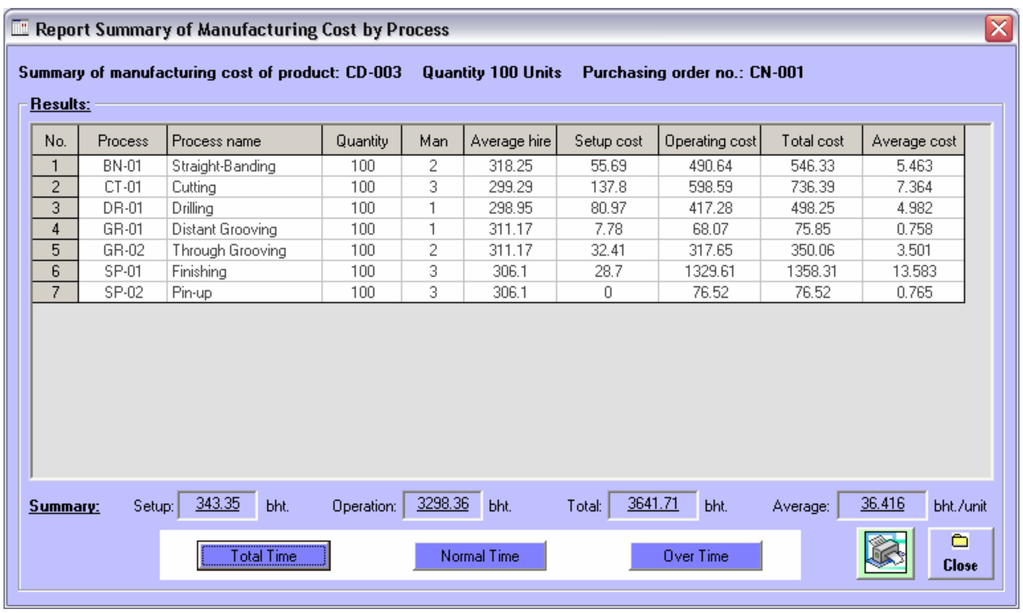
Results:

No.	Part code	Part name	Quantity	Man	Setup cost	Operating cost	Total cost	Average cost
1	CD-003/01	Left-Side Plate	100	2	101.53	521.74	623.27	6.233
2	CD-003/02	Right-Side Plate	100	2	25.18	470.76	495.94	4.959
3	CD-003/03	Vertical Fixed Plate	100	2	53.81	498.85	552.66	5.527
4	CD-003/04	Back Plate	100	3	22.45	65.47	87.92	0.879
5	CD-003/05	Top Plate	100	2	43.31	491.45	534.76	5.348
6	CD-003/06	Horizontal Fixed Plate	100	2	44.86	483.02	527.87	5.279
7	CD-003/07	Bottom Plate	100	2	37.26	358.24	395.5	3.955
8	CD-003/08	Base Plate	100	2	14.96	408.83	423.79	4.238

Summary: Setup: 343.36 bht. Operation: 3298.36 bht. Total: 3641.72 bht. Average: 36.42 bht./unit

Total Time | Normal Time | Over Time | Close

Figure 7.64 Summary report of manufacturing cost represented by part



Report Summary of Manufacturing Cost by Process

Summary of manufacturing cost of product: CD-003 Quantity 100 Units Purchasing order no.: CN-001

Results:

No.	Process	Process name	Quantity	Man	Average hire	Setup cost	Operating cost	Total cost	Average cost
1	BN-01	Straight-Banding	100	2	318.25	55.69	490.64	546.33	5.463
2	CT-01	Cutting	100	3	299.29	137.8	598.59	736.39	7.364
3	DR-01	Drilling	100	1	298.95	80.97	417.28	498.25	4.982
4	GR-01	Distant Grooving	100	1	311.17	7.78	68.07	75.85	0.758
5	GR-02	Through Grooving	100	2	311.17	32.41	317.65	350.06	3.501
6	SP-01	Finishing	100	3	306.1	28.7	1329.61	1358.31	13.583
7	SP-02	Pin-up	100	3	306.1	0	76.52	76.52	0.765

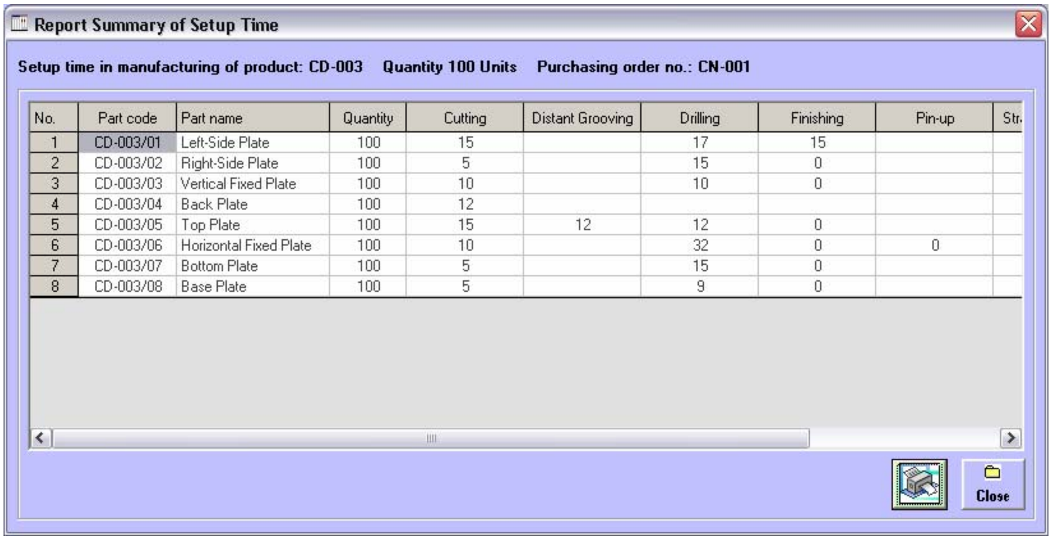
Summary: Setup: 343.35 bht. Operation: 3298.36 bht. Total: 3641.71 bht. Average: 36.416 bht./unit

Total Time | Normal Time | Over Time | Close

Figure 7.65 Summary report of manufacturing cost represented by process

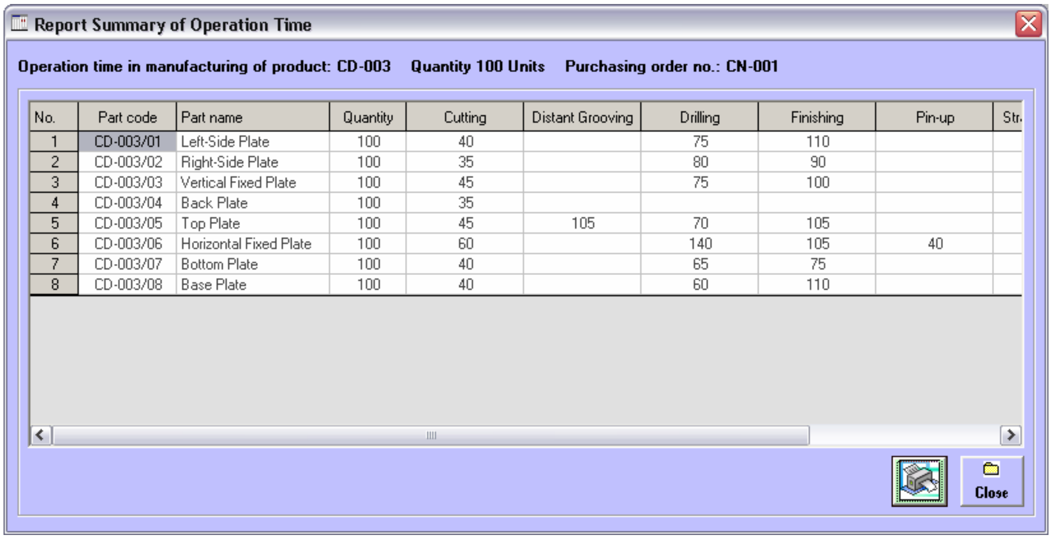
Additional reports

This group of reports represents the additional reports that are not included in the previous two groups. It represents the overview of manufacturing information of the selected item represented as crosstab between the parts and the processes i.e. the summary of setup time as represented in Figure 7.66, the summary of operation time as represented in Figure 7.67, the summary of man-hours, which means the manufacturing time of the work piece multiplies by manpower, as represented in Figure 7.68, and the summary of manpower as represented in Figure 7.69. The user can also print out these reports. The examples of printouts of these reports are represented in Annex II.



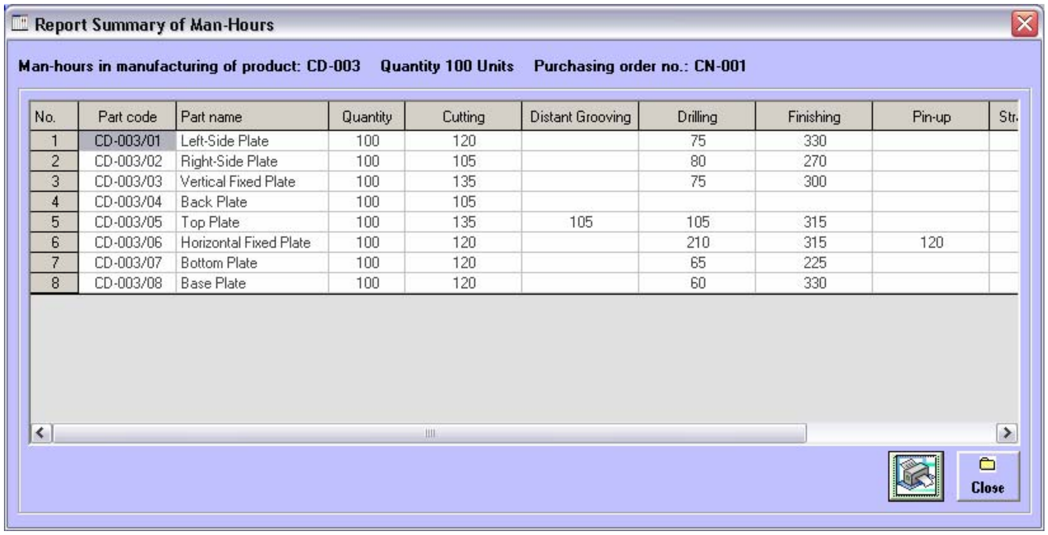
No.	Part code	Part name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Str.
1	CD-003/01	Left-Side Plate	100	15		17	15		
2	CD-003/02	Right-Side Plate	100	5		15	0		
3	CD-003/03	Vertical Fixed Plate	100	10		10	0		
4	CD-003/04	Back Plate	100	12					
5	CD-003/05	Top Plate	100	15	12	12	0		
6	CD-003/06	Horizontal Fixed Plate	100	10		32	0	0	
7	CD-003/07	Bottom Plate	100	5		15	0		
8	CD-003/08	Base Plate	100	5		9	0		

Figure 7.66 Crosstab summary report of setup time



No.	Part code	Part name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Str.
1	CD-003/01	Left-Side Plate	100	40		75	110		
2	CD-003/02	Right-Side Plate	100	35		80	90		
3	CD-003/03	Vertical Fixed Plate	100	45		75	100		
4	CD-003/04	Back Plate	100	35					
5	CD-003/05	Top Plate	100	45	105	70	105		
6	CD-003/06	Horizontal Fixed Plate	100	60		140	105	40	
7	CD-003/07	Bottom Plate	100	40		65	75		
8	CD-003/08	Base Plate	100	40		60	110		

Figure 7.67 Crosstab summary report of operation time

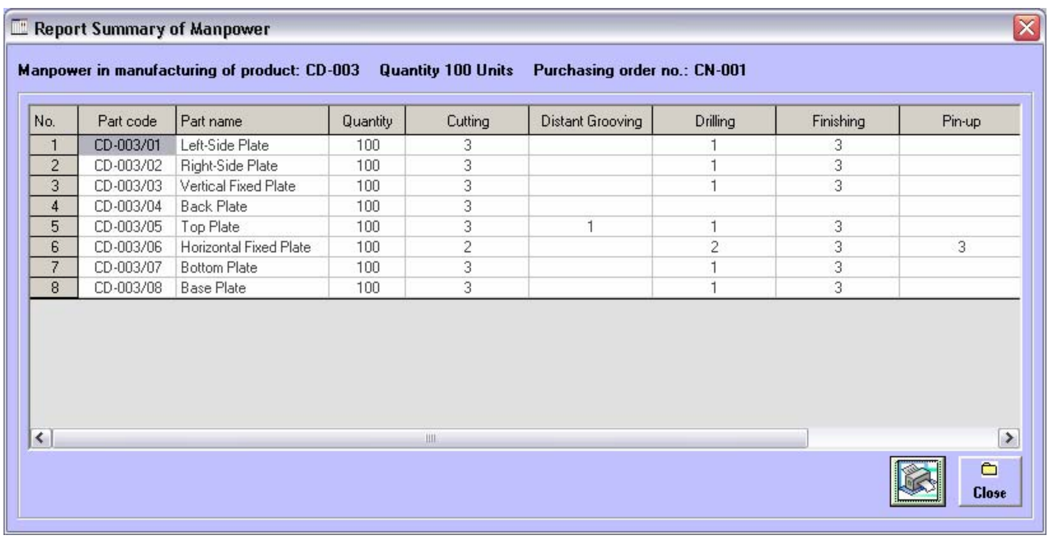

Report Summary of Man-Hours

Man-hours in manufacturing of product: CD-003 Quantity 100 Units Purchasing order no.: CN-001

No.	Part code	Part name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Str
1	CD-003/01	Left-Side Plate	100	120		75	330		
2	CD-003/02	Right-Side Plate	100	105		80	270		
3	CD-003/03	Vertical Fixed Plate	100	135		75	300		
4	CD-003/04	Back Plate	100	105					
5	CD-003/05	Top Plate	100	135	105	105	315		
6	CD-003/06	Horizontal Fixed Plate	100	120		210	315	120	
7	CD-003/07	Bottom Plate	100	120		65	225		
8	CD-003/08	Base Plate	100	120		60	330		

Close

Figure 7.68 Crosstab summary report of man-hours


Report Summary of Manpower

Manpower in manufacturing of product: CD-003 Quantity 100 Units Purchasing order no.: CN-001

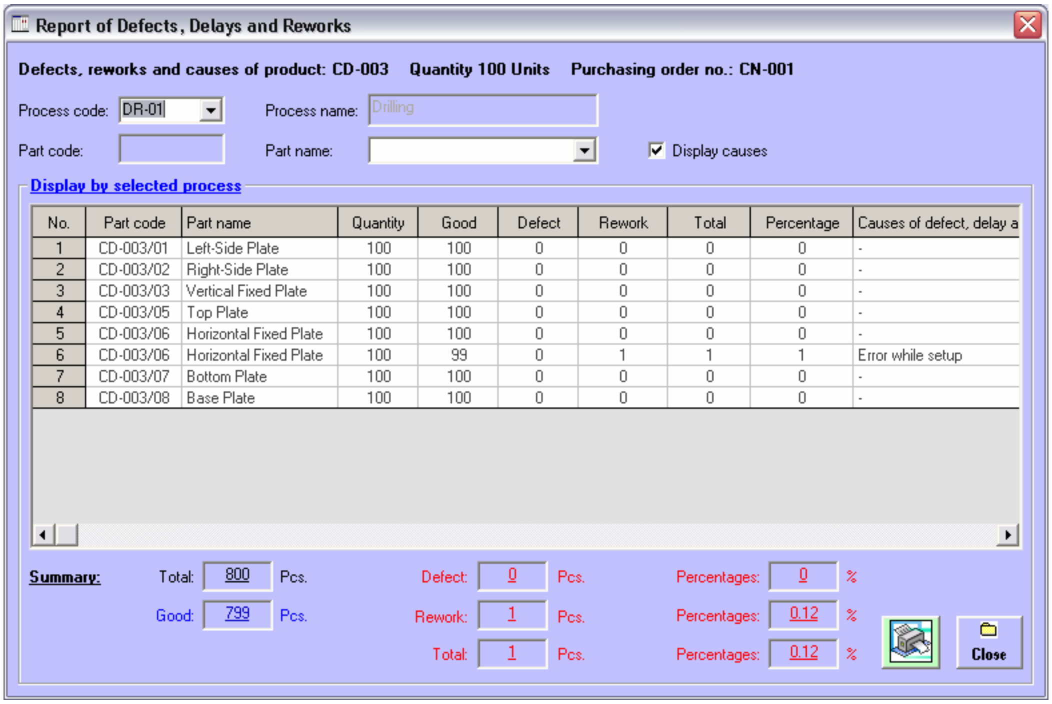
No.	Part code	Part name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up
1	CD-003/01	Left-Side Plate	100	3		1	3	
2	CD-003/02	Right-Side Plate	100	3		1	3	
3	CD-003/03	Vertical Fixed Plate	100	3		1	3	
4	CD-003/04	Back Plate	100	3				
5	CD-003/05	Top Plate	100	3	1	1	3	
6	CD-003/06	Horizontal Fixed Plate	100	2		2	3	3
7	CD-003/07	Bottom Plate	100	3		1	3	
8	CD-003/08	Base Plate	100	3		1	3	

Close

Figure 7.69 Crosstab summary report of manpower

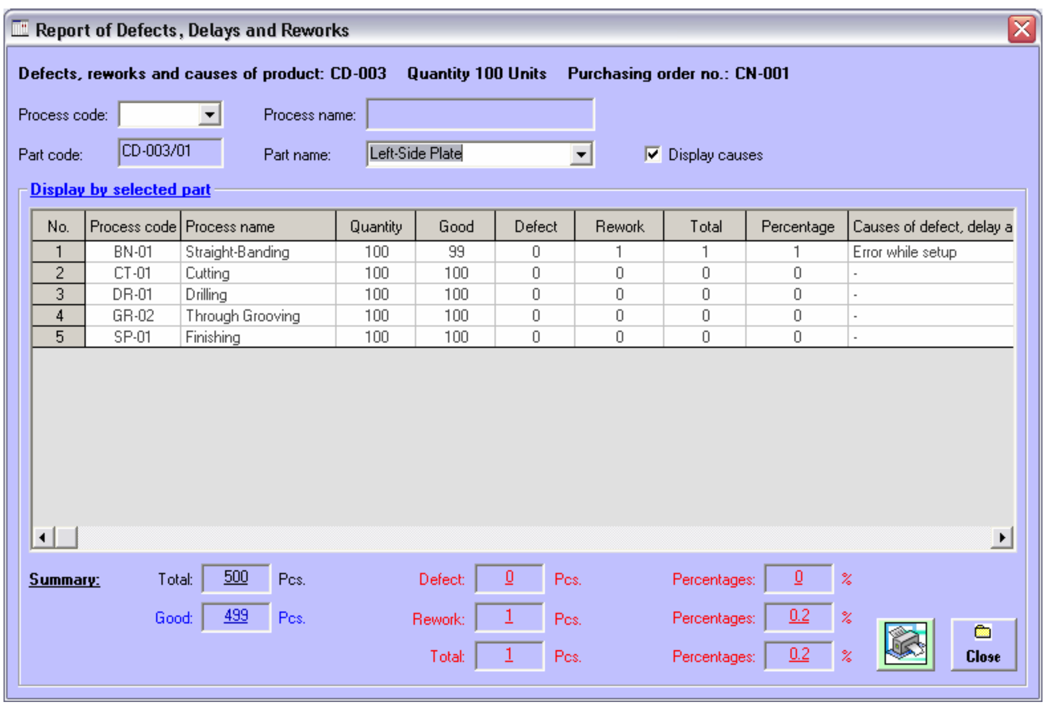
<u>Reports of defects and reworks</u>

These reports sum up the quantity of the produced parts, good parts, defects, reworks that are represented in both of number and percentage. It also gives the cause of defects, delays, and/or reworks. These reports can be represented either following the parts, as represented in Figure 7.70, or following the processes, as represented in Figure 7.71. The user can also print out these reports. The examples of printouts of these reports are represented in Annex II.



No.	Part code	Part name	Quantity	Good	Defect	Rework	Total	Percentage	Causes of defect, delay a
1	CD-003/01	Left-Side Plate	100	100	0	0	0	0	-
2	CD-003/02	Right-Side Plate	100	100	0	0	0	0	-
3	CD-003/03	Vertical Fixed Plate	100	100	0	0	0	0	-
4	CD-003/05	Top Plate	100	100	0	0	0	0	-
5	CD-003/06	Horizontal Fixed Plate	100	100	0	0	0	0	-
6	CD-003/06	Horizontal Fixed Plate	100	99	0	1	1	1	Error while setup
7	CD-003/07	Bottom Plate	100	100	0	0	0	0	-
8	CD-003/08	Base Plate	100	100	0	0	0	0	-

Figure 7.70 Summary report of defects and reworks represented by part



No.	Process code	Process name	Quantity	Good	Defect	Rework	Total	Percentage	Causes of defect, delay a
1	BN-01	Straight-Banding	100	99	0	1	1	1	Error while setup
2	CT-01	Cutting	100	100	0	0	0	0	-
3	DR-01	Drilling	100	100	0	0	0	0	-
4	GR-02	Through Grooving	100	100	0	0	0	0	-
5	SP-01	Finishing	100	100	0	0	0	0	-

Figure 7.71 Summary report of defects and reworks represented by process

7.5.4 Initialization of DAPP

According to Figure 7.32, we have divided the structure of DAPP into two parts: *Input information* and *Output information* as presented in section 7.5.2 and 7.5.3 respectively. DAPP stores the input information and some of output information into a relational database. To evaluate a design, the manufacturer needs firstly the person in the domain of enterprise to contribute fundamental information into the database of DAPP. We can categorize such information into departments followed the structure of DAPP as following:

- Sales & Marketing comprises *Customer Data* and *Purchasing Order*, as presented in section 7.5.2. This information supports the input data of *Daily Report*, and output information of *Production Order*, *Job Order*, and *Production Report*.
- Purchasing & Stock comprises *Raw materials* i.e. PB-MDF, Band, Foil, and *Purchased part* as presented in section 7.5.2. This information supports the input information in *Design* section for defining the materials using for each part, and supports the output information of material requirements.
- Engineering comprises *Work center*, *Machine*, and *Standard process* as presented in section 7.5.2. This information supports the input data of *Process Route Library and Daily Report*.
- Human Resource comprises *Employee Data* as presented in section 7.5.2. This information supports the output of *Cost Estimation*.

We can illustrate the structure of the initial information as represented in emphatic letters in Figure 7.72. Such *initial information* is required for evaluation of general design problems, and can be modified or appended if needed.

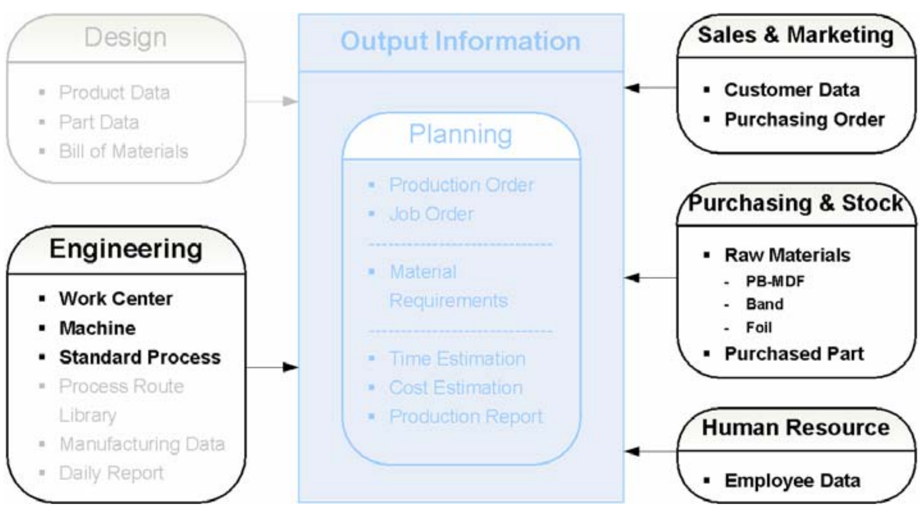


Figure 7.72 Structure of initial information of DAPP

7.5.5 Interaction between CoDeMo and DAPP

We have presented in the previous section about the initial information. To perform a design evaluation, the manufacturer needs the design team to propose or to contribute information of the design problem. To perform that task, the manufacturer asks CoDeMo to output the pertinent information in a neutral file, as presented before in Figure 7.29. As well, the manufacturer has to contribute results after achieved the evaluation. Figure 7.73 presents the interaction between CoDeMo and DAPP.

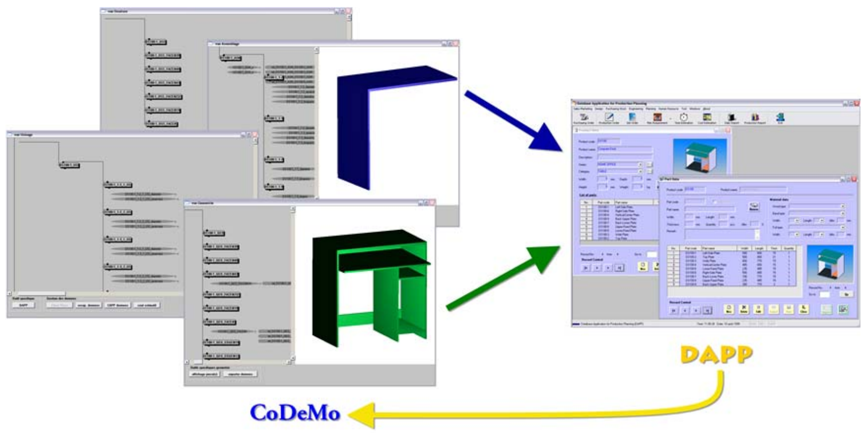


Figure 7.73 Interaction between DAPP and CoDeMo

CoDeMo outputs information of one part into one neutral file. The information in such neutral file concerns mainly the *Design* section. We present here, by example, an output file of the part *DS100-1* and the interaction between DAPP and CoDeMo. Table 7.7 presents a part of data in the output file of *DS100-1* which is the proposition of dimension and data of raw materials.

Table 7.7 Data presented in an output file concerning Part Data interface

PART: DS100-1

Details de forme:
surface_normale: x
taille_x: [15.000000 .. 15.000000]
taille_y: [800.000000 .. 800.000000]
taille_z: [500.000000 .. 500.000000]

Details de surfaces:
tolerance_largeur: [0.500000..0.500000]
tolerance_longueur: [0.500000..0.500000]
materiau: PVC
couleur: BEECH

Details de mecanique:
materiau: PB

The manufacturer retrieves this information into the database of DAPP via the interface of *Part Data* by click on the  button, and then chooses a corresponding file. Note that DAPP does not translate automatically all data presented in Figure 7.36. We still need competences of the manufacturer to complete data such allowance values of board, edge-bands, and foil utilization. Such values are depending on the product model, efficiency of machines, labor skills, and production system. Nevertheless, the manufacturer may use statistical information if it exists. The data presented in the interface of *Part Data* is the core data for further using in input information and output information. In addition, it is also used to estimate the requirement of raw materials. Figure 7.74 represents the result of exchanged data in *Part Data* interface by the manufacturer.

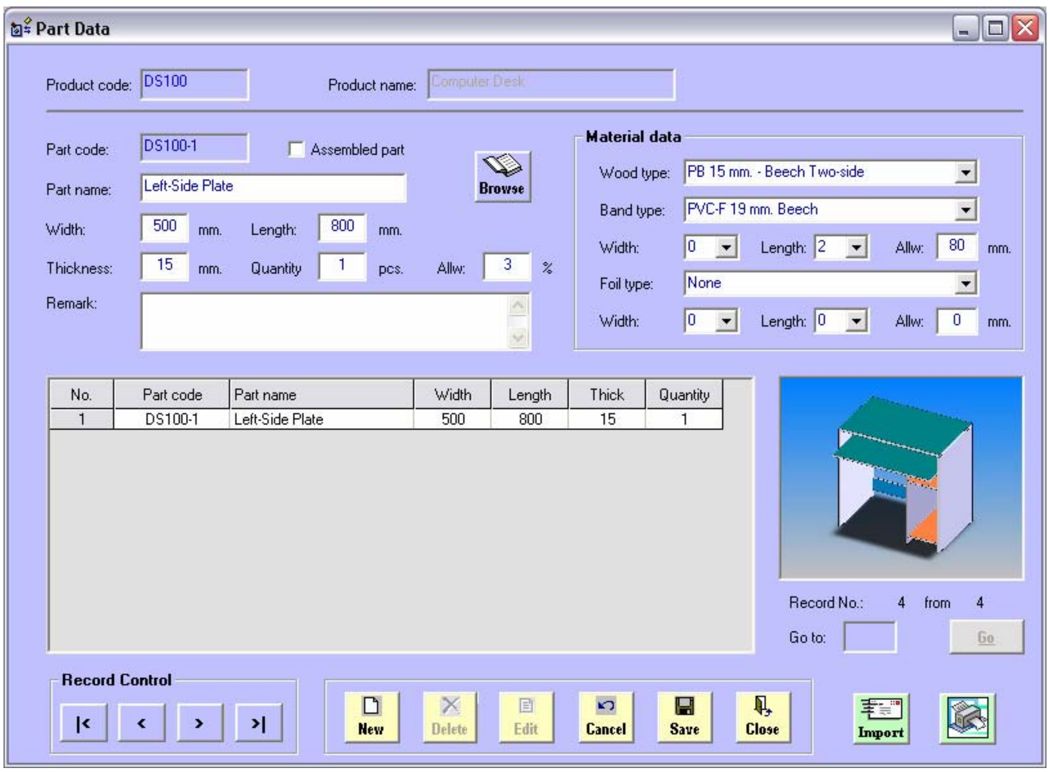


Figure 7.74 Exchanged data represented in Part Data interface

Table 7.8 presents another part of data in the output file of *DS100-1* which is the proposition of materials. This data enhances the manufacturer to identify a bill of materials for the product. However, this data concerns only the fasteners used in the product. In fact, a bill of materials is a list of purchases parts that also includes materials which may not a fastener. Therefore, the manufacturer or the person who is in charge is required to complete this data. An example of bill of materials has already represented in Figure 7.37.

Table 7.8 Data presented in an output file concerning assembly data

PART: DS100-1

Details d'assemblage:
surface: DS100-1_ASM_y
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-2_ASM
type contrainte: TourillonTraversanteCame
diametre_boitier 12 mm.; epaisseur_boitier 10 mm.;
diametre_boulon 8 mm.; longueur_boulon 48 mm.;
quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-3_ASM
type contrainte: TourillonVisseeCame
diametre_boitier 20 mm.; epaisseur_boitier 12.7 mm.;
diametre_boulon 8 mm.; longueur_boulon 18 mm.;
quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-7_ASM
type contrainte: Tourillon
diametre 8 mm.; longueur 36 mm.;
type Groove; quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-9_ASM
type contrainte: Rainure
largeur 4 mm.; profondeur 5 mm.; distance 280 mm.;

Table 7.9 presents another part of data in the output file of *DS100-1* which concerns the manufacturing process of the part. According to the chosen assembly solutions of the assembler and the production rules between assembly view and manufacturing view, CoDeMo outputs manufacturing data which enhances the manufacturer to complete the information in the *Manufacturing Data* interface.

Table 7.9 Data presented in an output file concerning manufacturing data

```
PART: DS100-1
        Details d'usinage:
        surface: DS100-1_ASM_y
        attribut associe: surface_normale
        relation associee: rel_DS100-1_ASM_DS100-2_ASM
        type contrainte: TourillonTraversanteCame
                percage1:   diameter 12 mm.;   profondeur 10.25 mm.;
                type:       non-debouchant;
                percage2:   diametre 8 mm.;    profondeur 34 mm.;
                type:       debouchant;

        surface: DS100-1_ASM_x
        attribut associe: surface_normale
        relation associee: rel_DS100-1_ASM_DS100-3_ASM
        type contrainte: TourillonVisseeCame
                percage1:   diametre 20 mm.;   profondeur 13 mm.;
                type:       non-debouchant;
                percage2:   diametre 8 mm.;    profondeur 10 mm.;
                type:       non-debouchant;

        surface: DS100-1_ASM_x
        attribut associe: surface_normale
        relation associee: rel_DS100-1_ASM_DS100-7_ASM
        type contrainte: Tourillon
                percage1:   diametre 8 mm.;    profondeur 24 mm.;
                type:       non-debouchant;
                percage2:   diametre 8 mm.;    profondeur 12 mm.;
                type:       non-debouchant;

        surface: DS100-1_ASM_x
        attribut associe: surface_normale
        relation associee: rel_DS100-1_ASM_DS100-9_ASM
        type contrainte: Rainure
                rainurage:  largeur 4 mm.;     profondeur 5 mm.; distance 280 mm.;
                type:       non-traversant;
```

We can summarize the input information contributed by CoDeMo into a form of the structure of DAPP as represented in emphatic letters in Figure 7.75. Note that the results of evaluation from DAPP must be traded into the shared database of CoDeMo. We will present an example of such exchanging of information in the next section.

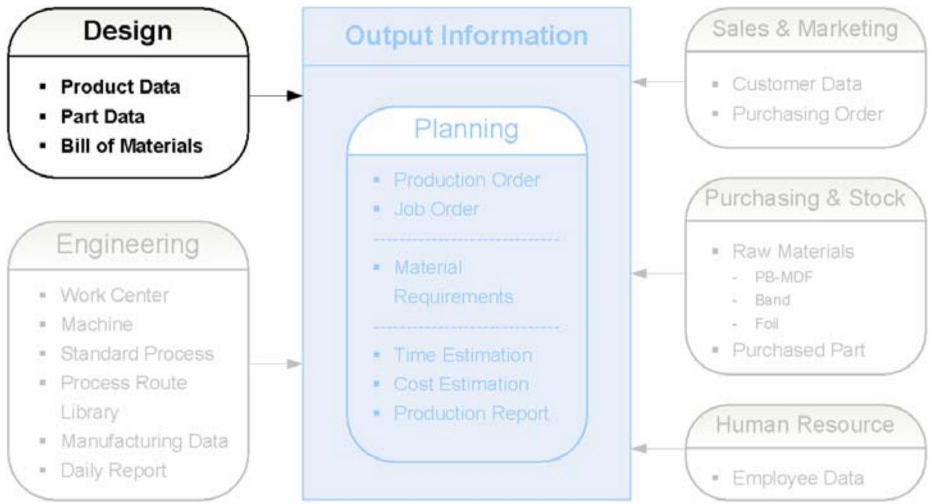


Figure 7.75 Structure of input information contributed by CoDeMo

The initial information in the previous section and the information contributed by CoDeMo in the previous section enhance the manufacturer to contribute his/her knowledge and experiences into the database of *DAPP*. The manufacturer is principally responsible to provide the information to the interface of *Process Route Library* and *Manufacturing Data*. *Process Route Library* is a sort of knowledge base that stores a list of routes of manufacturing processes, as presented in section 7.5.2. *Manufacturing Data* comprises the data of setup time, operation time, manpower, and descriptions, which needs to be complete by the manufacturer. The information in the *Part Data* interface and the manufacturing data in the output file from CoDeMo imply some manufacturing data for the manufacturer. Together with the knowledge and experiences, the manufacturer can complete the required information. The manufacturing data such setup time and operation time may be complete by using the well-known method of time and study [Taylor 1911] for calculating standard time, or the analysis of statistical data if it exists.

With the input information, the manufacturer can evaluate the proposed design problems. The manufacturer can receive continually information from CoDeMo while evaluating the design as parallel functioning. For example, when the assembler has completely defined the characteristics of an assembly solution for one part, the manufacturer can define consequently a process route for that part and its manufacturing data. This process continues until the evaluation is achieved. In addition, to monitor the manufacturing process, DAPP requires the manufacturer or the person who is in charge of this domain to complete the data in *Daily Report* interface. This statistical data permits the manufacturer or the production planner to analyze the results of process planning and then to improve the production schedule following the real situation. We can summarize the input information contributed by the manufacturer and the output information contributed by DAPP into a form of the structure of DAPP as represented in emphatic letters in Figure 7.76.

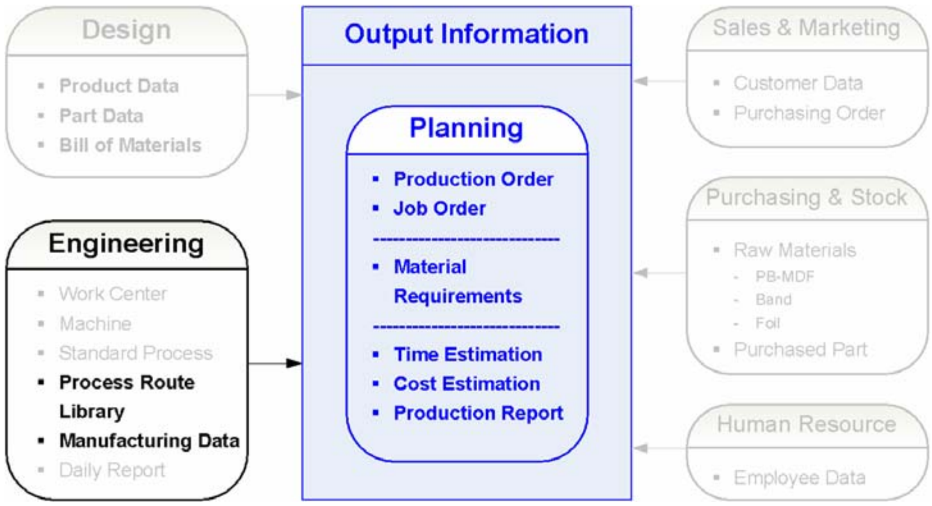


Figure 7.76 Structure of input information contributed by the manufacturer and output information

As soon as the design evaluation is achieved, DAPP contributes the results, e.g. material requirement, manufacturing cost (setup and operation), raw material and purchased part cost, manufacturing time, production reports etc., as represented in the 'Output Information' frame of Figure 7.76. At this time, the manufacturer is asked to provide the corresponding information to the shared database. To achieve that task, DAPP outputs the results into a text file which the internal actor is familiar with. Actually, in the manufacturing view, the manufacturer permits the internal actor to translate the results into the shared database of CoDeMo by click on the cout estimatif button. Table 7.10 presents an example of a text file that stores the result of cost estimation, see also Figure 7.56. Figure 7.77 presents, by example, the values of corresponding characteristics of instances after the translation process.

Table 7.10 Example of output information from DAPP

```
PRODUCT
    product: DC100
    rawmat_cost: 470.43
    purchased_part_cost: 108.88
    setup_cost: 203.34
    operation_cost: 27.28
    total_cost: 809.93

PART
    mpart_code: DC100-01
    rawmat_cost: 76.60
    operation_cost: 3.74
    mpart_code: DC100-02
    rawmat_cost: 76.60
    operation_cost: 3.74
    mpart_code: DC100-03
    rawmat_cost: 50.73
    operation_cost: 3.16
    mpart_code: DC100-04
    rawmat_cost: 6.11
    operation_cost: 0.56
    mpart_code: DC100-05
    rawmat_cost: 21.49
    operation_cost: 2.97
    mpart_code: DC100-06
    rawmat_cost: 19.09
    operation_cost: 2.39
    mpart_code: DC100-07
    rawmat_cost: 19.09
    operation_cost: 2.39
    mpart_code: DC100-08
    rawmat_cost: 71.12
    operation_cost: 3.73
    mpart_code: DC100-09
    rawmat_cost: 129.60
    operation_cost: 4.60
@
```

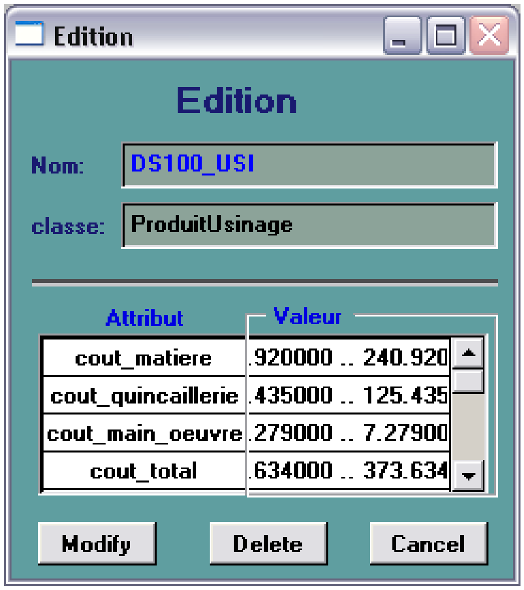


Figure 7.77 Values of the corresponding characteristics after the translation

7.6 Summary

This chapter has presented the validation of using the integrated design system, CoDeMo. We apply CoDeMo to create a collaborative environment and to bring different design actors into a virtual meeting room. The integrated design process has begun by the technologist who initiated the translation process. The internal actor translates a STEP AP-203 file, which is a conceptual model of the product, into CoDeMo. Other design actors consequently contribute their data and constraints as soon as they have enough information to justify and to evaluate the design or a part of the design. The design actors have constituted knowledge model and constraints by defining production rules in a neutral file for sharing/exchanging information via GUIs contributed by CoDeMo. The design actors can apply their specific application to solve design problems and then give the results into the design process. Nevertheless, we require further a common space for storing and for sharing the documents from any design actors, which are created during the design process cf. IPPOP project. In order to exchange information and results of the evaluations between CoDeMo and the specific applications, we have developed both CoDeMo and the specific applications to recognize the exchanged format of each other. We also have presented a specific application, DAPP, using in manufacturing view. It has presented that one design actor can introduce his/her tool to CoDeMo without a problem.

Conclusion

The objectives of this study were to develop a design modeller for an integrated design system, to propose a method to reduce the imaginary complexity in the design process, and to develop a design process for the industry of wood furniture made of particleboard and medium-density fiberboard.

We have mentioned about the globalization, which is the consequence of the evolution of information technology and communication. The globalization has challenged the design team to develop the design process to satisfy the customer's requirements with given criteria of quality, cost, time, and recycling.

In this study, we benefit the information technology in developing a design modeller for an integrated design system. A cooperative design modeller (CoDeMo) has been proposed. CoDeMo is based on a client-server system. We develop CoDeMo to create a collaborative environment that brings the design actors into a virtual meeting room in order to perform the design activities together with distant and synchronous or asynchronous access. Different from general CAD systems, CoDeMo is an integrated design system that does not take into account only the geometrical data, but it supports the aspects from different design actors from different domain of competences. With the developed methods and models for integration, CoDeMo permits the design actors to share and to exchange their information during the design process. We have applied the concept of product model to facilitate the design actors to construct the product structure and to store the product data in the form of data model and knowledge model.

One main objective of the integrated design is to reduce the design iterations by taking into account constraints from different disciplines as soon as possible before making a decision. To do so, the design actors have to contribute their information to the design team. The concept multi-representation allows the design actors to present their information into the collaborative environment while the concept of multidisciplinary, multi-actor and multi-view, permits the design actors to dialog, to discuss, to negotiate and to compromise during the design process. In addition, to manage knowledge and to keep up-to-date of information from different actors, the methods of data propagation and data translation are applied in this study.

Due to the growing demand of customization, the aspect of producers is to satisfy as much as possible the customer's requirements for having advantage in competitiveness. This aspect increases inevitably the complexity in the design process.

According to Axiom Design, Nam Suh states that the time-independent imaginary complexity can occur when we must satisfy many function requirements at the same time. In addition, this sort of complexity rises due to lacking of information of design actors. A simple solution of reducing such complexity is that to make the designers know what they should know.

To resolve such problem of complexity, we suppose that the design problem is an imaginary complexity as a hypothesis in this study. Consequently, we propose the integrated design system, CoDeMo, to picture the design actors into a virtual meeting room. We postulate that each design actor has knowledge and experiences on design problems and may have an access to the existing data of the design problems. We also postulate that design actors work in the notion of “just need” during they are solving the design problems in the design process. With the support of method and models for integration, the integrated design system enhances the design team to reduce the complexity by solving the problems of uncoupled design, decoupled design, and weak coupled design.

In this study, we have divided the integrated design process into two phases. The first design phase is mainly concerned by the technologist. S/he has to set off the initial information by transforming the conceptual design of a product, which is handled by a CAD system, into the collaborative environment. This initial information comprises global form and dimension of the conceptual product, which is the starting point for the integration of knowledge. The task of the technologist is to accomplish the functional surface of the conceptual product. In the second design phase, other design actors are asked to participate in the design process to contribute their information, constraints, and points of view to the design team. In this study, we have taken into account principally three domains of competence: assembly, mechanic, and manufacturing. The task of the assembler is to choose appropriated assembly solutions to the product; the mechanician has to define appropriate materials and thickness for each part while the manufacturing must evaluate manufacturing process and cost. Each design actor has different tasks but their information is relative to each other.

To introduce such information into the product, we have applied the concept of features and production rules into this study. We have used features to describe characteristics and behaviors of the product. According to the concept of “worlds of design”, we have classified features into three significations: vernacular, vehicular, and universal. The system permits the design actors to create and to use such features by giving an access to a feature based engine. To keep coherence of the contributed information, the concept of production rules is applied. A production rule is an element of knowledge which is used in problem-solving process. It enhances the design actors to share and to exchange their information to the team. Production rules are stored in the system in a form of neutral file, named *QTrans*. To permit the design actors to create and to use such temporal knowledge, the system must give them an access to an inference engine.

We have applied the integrated design system to the industry of wood furniture made of particle board and medium-density fiber board. Due to the short life cycle and rapid changing of models and styles of this sort of furniture, it is difficult to manage the manufacturing process only by an integrated design modeller. Therefore, the integrated design system permits the design actors to employ their specific applications to evaluate the design. In this study, we have introduced a specific application for the manufacturing view, DAPP. This application uses a relational database for storing manufacturing data which is contributed by the manufacturer himself, other trade views, and CoDeMo. To exchange data between CoDeMo and DAPP, we have developed DAPP to be able to retrieve some provided data from CoDeMo, which is stored in a neutral file. According to the competence and experiences of the manufacturer, such information enhances him to evaluate the manufacturing process, to estimate manufacturing time and cost for each part of product. To contribute results of the evaluation to the shared database, DAPP outputs those results into a text file that the internal actor is familiar with.

Towards the end, the results which are contributed from the design team will be defined to product by features. If there is any unknown value, it means that the design process has not finished yet. The design is achieved when all values of characteristics of instances have been defined and accepted from the design team so the design satisfies the functional requirements.

Perspectives

We have succeeded in applying the integrate design system, CoDeMo, to the design process of the wood furniture made of particleboard and medium-density fiberboard. Using of CoDeMo enhances the design team to reduce the time-independent imaginary complexity in the design process. Nevertheless, one may notice that the system may create some design iterations although the design actors work in the notion of "just need". If the proposed solution does not satisfy FRs, the design actor(s), who are concerned, will be asked to re-discuss on the problem and to negotiate until they find the accepted one. This source of design iterations occurs due to single evaluation of design actors. In this situation, we project to develop further a concept of '*multi-variant*'. This method must permit the design actors to be able to propose to the team more than one solution for one problem. As a result, the design team could reduce a number of re-design processes and consequently the time consuming of the design process.

CoDeMo permits the design actors to use their specific applications for evaluating the design. In the mechanical view, we have applied a specific application, RDM 6, for evaluating the deflection of plates, which occurs by a vertical static load. Nevertheless, some standards may require some other tests such vertical impact test, drop test, etc. Therefore, this trade view may require specific applications for such tests or even require to be developed to evaluate such tests.

In the manufacturing view, we have applied a specific application for the manufacturer, DAPP, for creating conceptual manufacturing process, and for estimating manufacturing time and cost. We project to add some modules for storing the information of available machines and tools. This information would support afterward the production planner to plan the manufacturing processes and to manage the schedule of machine operations. In addition, one may notice that DAPP requires quite a lot of information for evaluating the design. In order to facilitate the manufacturer in acquisition of such information, we may apply a concept of case-based reasoning system or the *generative* method to recognize similar parts in the database. On the other hand, the *variant* method or even the *hybrid* method [Lutters et al 2006] may be required. Furthermore, the use of manufacturing features may be required in manufacturing process planning [Martin and Meausoone 1999], [Meausoone and Martin 2001]. These methods would enhance DAPP to estimate materials requirement of a product, and also to estimate its manufacturing time and cost.

Bibliography

[APA 2005] APA, *The Plywood Age*: APA - The Engineered Wood Association, 2005, pp. 175, URL: http://www.apawood.org/plywoodpioneers/plyage.htm.

[Archer 1973] B. Archer, "The Need for Design Education," Royal College of Art, 1973. Cited in [UK Technology Education Centre 1996], "What is Design?," 1996, URL: http://atschool.eduweb.co.uk/trinity/watdes.html.

[ASTM 2005] ASTM, "Standard Terminology Relating to Wood-Base Fiber and Particle Panel Materials," in *ASTM Book of Standards*, vol. 04.10: ASTM International, 2005, pp. 4.

[Belloy 1994] P. Belloy, "Intégration des connaissances métier dans la conception: un modèle pour les pièces mécaniques. Application à l'usinage et à l'estampage," in *Mécanique*. Saint-Martin-d'Hères, FRANCE: Université Joseph Fourier de Grenoble, 1994, pp. 192.

[Bodig and Jayne 1982] J. Bodig and B. A. Jayne, *Mechanics of Wood and Wood Composites*. New York: Van Nostrand Reinhold, 1982, ISBN: 0-442-00822-8.

[Boujut and Tichkiewitch 1995] J.-F. Boujut and S. Tichkiewitch, "A Co-ordination Tool in the Design Process of Stamped Parts," presented at AMPT '95, Advances in Materials and Processing Technologies, Dublin, 1995.

[Brenden and Schaffer 1980] J. J. Brenden and E. L. Schaffer, "Smoldering wave-front velocity in fiberboard," Madison, WI: U.S. Department of Agriculture, Forest Service, Forest Products Laboratory, 1980, pp. 6.

[Brissaud et al 1997] D. Brissaud, H. Paris, and S. Tichkiewitch, "Assisting designers in the forecasting of surfaces used for easier fixturing in the machining process," *Journal of Materials Processing technology*, vol. 65, pp. 26-33, 1997.

[Brissaud and Tichkiewitch 2000] D. Brissaud and S. Tichkiewitch, "Innovation and manufacturability analysis in an integrated design context," *Computers in Industry*, vol. 43, pp. 111-121, 2000.

[Butdee 2002] S. Butdee, "Integration of product and process design for wood furniture industry," presented at Seventh International Pacific Conference on Manufacturing and Management, Bangkok, Thailand, 2002.

[Canadian Forest Industries 2006] Canadian Forest Industries, "Wood-Based Panel Products: Technology Roadmap," in *Innovation, Research and Technology*, vol. 2007: PWGSC: Public Works and Government Services Canada, 2006.

[Chapa Kasusky 1997] E. C. Chapa Kasusky, "Outils et structure pour la coopération formelle et informelle dans un contexte de conception holonique," in *Mécanique: Conception, géomécanique et matériaux*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 1997, pp. 132.

[Clausing 1994] D. P. Clausing, *Total quality development: a step-by-step guide to world-class concurrent engineering*. New York: ASME, 1994.

[De Turck 2005] B. de Turck, "The competitiveness of the furniture industry," presented at The Future of the Furniture Industry in an Enlarged Europe, Brussels, Belgium, 2005.

[Debard 2000a] Y. Debard, "RDM 6," 6.07 ed. Le Mans, 2000.

[Debard 2000b] Y. Debard, *Découvrir le calcul de structure avec RDM 5*. Le Mans, 2000, ISBN: 2-9511025-1.

[Domb 1997] E. Domb, "QFD and TIPS/TRIZ," presented at The Third Annual International QFD Symposium, Linköping, Sweden, 1997.

[EERE 2005] EERE, "Integrated Building Design," in *Building Technologies Program*, vol. 2007: U.S. Department of Energy, 2005.

[English 1994] B. English, J. A. Youngquist, and A. M. Krzysik, "Lignocellulosic Composites," in *Cellulosic Polymers, Blends and Composites*, R. D. Gilbert, Ed. New York: Hanser Publishers, 1994, pp. 115–130.

[Erens 1996] F.-J. Erens, "The Synthesis of Variety: Developing Product Families," in *Department of Technology Management*. Eindhoven, Netherlands: Eindhoven University of Technology, 1996, pp. 309.

[Feng and Song 2000] S. C. Feng and E. Y. Song, "Information Modeling of Conceptual Process Planning Integrated with Conceptual Design," presented at The 2000 ASME Design Engineering Technical, Baltimore, Maryland, 2000.

[FFE 2006] FFE, "International: A Rosy Picture for 2006," in *Furniture & Furnishing Export International*, 2006, URL: http://www.apsmediagroup.com.

[Gaucheron 2000] T. Gaucheron, "Intégration du recyclage en conception Le Modèle produit : Un outil descriptif et cognitif dans le processus de prise en compte du recyclage," in *Génie Industriel*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 2000, pp. 252.

[Gershenson and Heylighen 2005] C. Gershenson and F. Heylighen, "How can we think the complex?," in *Managing Organizational Complexity: Philosophy, Theory and Application*, vol. 1, *Managing the Complex*, K. Richardson, Ed.: Information Age Publishing, 2005, pp. 47-62.

[Gruber 1993] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," *Knowledge Acquisition*, vol. 2, pp.199--220, 1993.

[Gzara Yesilbas et al 2006] L. Gzara Yesilbas, B. Rose, and M. Lombard, "Specification of a repository to support collaborative knowledge exchanges in IPPOP project," *Computers in Industry*, vol. 57, pp. 690 - 710, 2006.

[Hayes 2006] M. Hayes, "Agricultural Residues: A Promising Alternative to Virgin Wood Fiber," 2006.

[Heylighen 2002] F. Heylighen, "Complexity and Information Overload in Society: why increasing efficiency leads to decreasing control," 2002, pp. 44.

[Huang 2004] C.-C. Huang, "A Multi-agent Approach to Collaborative Design of Modular Products," *Concurrent Engineering*, vol. 12, pp. 39-47, 2004.

[ILOG] ILOG, Bâtiment ORSUD, 3-5 avenue Galiéni 94257 Gentilly CEDEX, France, URL: http://www.ilog.com/.

[Kara et al 1999] S. Kara, B. Kayis, and H. Kaebernick, "Modelling Concurrent Engineering Projects Under Uncertainty," *Concurrent Engineering*, vol. 7, pp. 269-274, 1999.

[Kirkpatrick and Barnes 2006] J. W. Kirkpatrick and H. M. Barnes, "Biocide treatments for wood composites--A review," 2006, pp. 21.

[Kvan 2000] T. Kvan, "Collaborative design: what is it?," *Automation in Construction*, vol. 9, pp. 409-415, 2000.

[Lee 2003] T. Lee, "Complexity Theory in Axiomatic Design," in *Mechanical Engineering*: Massachusetts Institute of Technology, 2003, pp. 182.

[Lombard et al 2005] M. Lombard, B. Rose, and G. Ris, "Design in collaboration : existing trends and application to the case of conflict handling with CO²MED software," presented at 16th IFAC World Congress, Prague, Czech Republic, 2005.

[Lu 2004] S. C.-Y. Lu, "Beyond Concurrent Engineering: A New Foundation for Collaborative Engineering," presented at CE2004: The 11th ISPE International Conference on Concurrent Engineering: Research and Applications, Beijing, China, 2004.

[Lu 2006a] S. C.-Y. Lu, "Complexity in Collaborative Engineering," presented at International Consortium for Complexity Research (ICCR) Forum, Alberta, Canada, 2006.

[Lu 2006b] S. C.-Y. Lu, "A Socio-Technical Foundation for Collaborative Engineering Design," 2006, URL: http://wisdom.usc.edu/iccr/pmwiki.php?n=Reading.Article4.

[Lu 2007] S. C.-Y. Lu, "A scientific foundation of collaborative engineering," will be presented at 57th General Assembly of CIRP, Dresden, Germany, 2007.

[Lutters et al 2006] E. Lutters, M. Toxopeus, and F. Van Houten, "Combined assessment methods for decision support in product development," presented at CIRP International Conference on Life Cycle Engineering, Leuven, Belgium, 2006.

[Lyons 1997] M. J. Lyons, "FAO Outlook study on wood based panels production, consumption and trade in the Asia Pacific region 1996 to 2010," Forestry Policy and Planning Division, Kuala Lumpur, Malaysia August 1997 1997.

[Maloney 1986] T. M. Maloney, "Terminology and products definitions - A suggested approach to uniformity worldwide," presented at 18th international union of forest research organization world congress, Ljubljana, Yugoslavia, 1986.

[Mann 1999a] D. Mann, "Axiomatic Design and TRIZ: Compatibilities and Contradictions," *The TRIZ Journal*, 1999, URL: http://www.triz-journal.com/archives/1999/06/a/index.htm.

[Mann 1999b] D. Mann, "Axiomatic Design and TRIZ: Compatibilities and Contradictions Part II," *The TRIZ Journal*, 1999, URL: http://www.triz-journal.com/archives/1999/07/f/index.htm.

[Martin and Meausoone 1999] P. Martin and P.-J. Meausoone, "Use of manufacturing features for Computer aided process planning and manufacturing system design, Application to Furnitures parts of solid wood Advanced manufacturing systems and technology," in *Advanced manufacturing systems and technology*, E. Kuljanic, Ed.: Springer-Verlag, 1999, pp. 281-288.

[Martin and A. D'Acunto 2003] P. Martin and A. D'Acunto, "Design of a production system: an application of integration product-process," *International Journal of Computer Integrated Manufacturing*, vol. 16, pp. 509 - 516, 2003.

[Mcmahon and Browne 1998] C. Mcmahon and J. Browne, *CADCAM: Principles, Practice and Manufacturing Management*, 2Rev Ed edition (Mar 1998) ed: Prentice Hall, 1998.

[Meausoone and Martin 2001] P.-J. Meausoone and P. Martin, "Génération automatique de processus pour les pièces de meubles en bois massif," *Journal Européen des Systèmes Automatisés*, vol. 35, pp. 1215-1234, 2001.

[Mer 1998] S. Mer, "Les mondes et les outils de la conception. Pour une approche socio-technique de la conception de produit," in *Génie Industriel*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 1998, pp. 230.

[Mer et al 1995] S. Mer, A. Jeantet, and S. Tichkiewitch, *Les objets intermédiaires de la conception : modélisation et communication*. Paris, FRANCE: Europia, 1995.

[Molina et al 2005] A. Molina, J. Aca, and P. Wright, "Global collaborative engineering environment for integrated product development," *International Journal of Computer Integrated Manufacturing*, vol. 18, pp. 635 - 651, 2005.

[Myint 1999] S. Myint, "Concurrent Engineering (CE) in Product Design: A Multicriteria Decision-Making Approach," *Concurrent Engineering*, vol. 7, pp. 2-9, 1999.

[Nakagawa 2007] T. Nakagawa, "A New Paradigm for Creative Problem Solving: Six-Box Scheme in USIT," 2007, URL: http://www.triz-journal.com/archives/2007/01/06/.

[NF Ameublement] NF Ameublement, "NF for Professional Furniture," CTBA, URL: http://www.nf-mobilier-pro.com/.

[Noël et al 2003] F. Noël, D. Brissaud, and S. Tichkiewitch, "Integrative Design Environment to Improve Collaboration between Various Experts," presented at 53rd General Assembly of CIRP, Montreal, Canada, 2003, pp.109-112.

[Noël et al 2004] F. Noël, L. Roucoules, and D. Teissandier, "Specification of product modelling concepts dedicated to information sharing in a collaborative design context," presented at 5th International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMME 2004, Bath, United Kingdom, 2004.

[Noël and Tichkiewitch 2004] F. Noël and S. Tichkiewitch, "Shared dynamic entities technology to support distant coordination in design activity," presented at 54th General Assembly of CIRP, Krakow, Poland, 2004, pp.163-166.

[Noël 2007] F. Noël, "A product - process - organisation integrative model to support collaborative design," in *Innovation in Life Cycle Engineering and Sustainable Development*, S. Netherlands, Ed., 2007, pp. 407-418, ISBN: 978-1-4020-4601-8.

[Pahl and Beitz 1996] G. Pahl and W. Beitz, *Engineering Design -- A Systematic Approach*, Second edition ed. London, Great Britain: Springer, 1996, ISBN: 3-540-19917-9.

[Parkinson et al 2000] B. Parkinson, P. Hudson, R. Senior, C. Short, and R. Barker, "A Virtual Environment for Concurrent Engineering using Artificial Intelligent Techniques," Innovative Design Engineering Research, 2000, URL: http://www.ider.herts.ac.uk/ider/DESCON.HTM.

[Pixler 1999] B. Pixler, "A Virtual History of Furniture," in *History and Philosophy of Vocational Education*, 1999, URL: http://www.usd382.k12.ks.us/pedigo/histfurn/.

[Poolton et al 2000] J. Poolton, H. S. Ismail, and S. M. M. Shahidipour, "The New Products Process: Effective Knowledge Capture and Utilisation," *Concurrent Engineering*, vol. 8, pp. 133-143, 2000.

[Prasad 1996] B. Prasad, *Concurrent Engineering Fundamentals: Integrated Product and Process Organization*, vol. 1. New Jersey: Prentice Hall PTR, 1996, ISBN: 0-131-47463-4.

[Pritchard 2002] D. J. Pritchard, "The Global Decentralization of Commercial Aircraft Production: Implications for U.S. Based Manufacturing Activity," in *Department of Geography*: University at Buffalo, 2002, pp. 115.

[Pugh 1991] S. Pugh, *Total Design: Integrated methods for successful product engineering*. Wokingham, England: Addison-Wesley, 1991, ISBN: 0-201-41639-5.

[Radulescu 2005] B. Radulescu, "Développement d'un système de contrôle et de gestion des processus de conception intégrée : application à la traduction et la substitution au sein d'un modèle produit," in *Mécanique: Conception, géomécanique et matériaux*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 2005, pp. 234.

[Roucoules 1999] L. Roucoules, "Méthodes et connaissances : contribution au développement d'un environnement de conception intégrée," in *Mécanique*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 1999, pp. 217.

[Roucoules and Tichkiewitch 2000] L. Roucoules and S. Tichkiewitch, "CoDE: A Cooperative Design Environment-- A New Generation of CAD Systems," *Concurrent Engineering*, vol. 8, pp. 263-280, 2000.

[Roucoules et al 2003] L. Roucoules, O. Salomons, and H. Paris, "Process planning as an integration of knowledge in the detailed design phase," *International Journal of Computer Integrated Manufacturing*, vol. 16, pp. 25-37, 2003.

[Roucoules et al 2006] L. Roucoules, F. Noël, D. Teissandier, M. Lombard, G. Débarbouillé, P. Girard, C. Merlo, and B. Eynard, "IPPOP: An opensource collaborative design platform to link product, design process and industrial organisation information," presented at 6th International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMME 2006, Grenoble, France, 2006.

[Salomone 1995] T. A. Salomone, *What every engineer should know about Concurrent Engineering*. New York: Marcel Dekker, 1995, ISBN: 0-8247-9578-4.

[Shpitalni and Lipson 1997] M. Shpitalni and H. Lipson, "Automatic Reasoning for Design under Geometrical Constraints," presented at 47th General Assembly of CIRP, Tianjin, China, 1997.

[Singh 1996] N. Singh, *Systems approach to Computer-integrated design and Manufacturing*. New York, USA: John Wiley & sons, 1996, ISBN: 0-471-58517-3.

[Sky and Buchal 1999] R. W. E. Sky and R. O. Buchal, "Modeling and Implementing Concurrent Engineering in a Virtual Collaborative Environment," *Concurrent Engineering*, vol. 7, pp. 279-289, 1999.

[SmarTeam] Dassault systems, URL: http://www.smarteam.com

[Smith and Wolcott 2006] P. M. Smith and M. P. Wolcott, "Opportunities for Wood/Natural Fiber-Plastic Composites in Residential and Industrial Applications," *Forest Products Journal*, vol. 56, pp. 4-11, 2006.

[Sohlenius 1992] G. Sohlenius, "Concurrent Engineering," presented at 42nd General Assembly of CIRP, Aix-en-Provence, France, 1992.

[Sohlenius 2004] G. Sohlenius, "Complexity, Difficulty and Axiomatic Design," N. P. Suh, Ed., 1996-2004, URL: http://wisdom.usc.edu/iccr/pmwiki.php?n=Main.NoteOnComplexityDifficultyAndAxiomaticDesign.

[Soltis 1999] L. A. Soltis, "Structural Analysis Equations," in *Wood handbook—Wood as an engineering material*, F. P. Laboratory, Ed., Rev. 1999 ed. Madison, Wisconsin: Madison, WI: U.S. Department of Agriculture, Forest Service, Forest Products Laboratory, 1999, pp. 8-1—8-11.

[Stagney 2003] D. B. Stagney, "The Integrated Concurrent Enterprise," in *Aeronautics and Astronautics*: Massachusetts Institute of Technology, 2003, pp. 180.

[Suh 1990] N. P. Suh, *The principles of Design*. New York: Oxford University Press, 1990, ISBN: 0-19-504345-6.

[Suh 2001] N. P. Suh, *Axiomatic Design Advances and Applications*. New York: Oxford University Press, 2001, ISBN: 0-19-513466-4.

[Suh 2003] N. P. Suh, "A Theory of Complexity and Applications," 2003, URL: http://wisdom.usc.edu/inspiring/resource/bioengineering/Namsuh.pdf.

[Suh 2005] N. P. Suh, "Complexity in engineering," presented at 55th general assembly of CIRP, Antalya, Turkey, 2005.

[Tatray 1992] P. Tatray, "Probabilistic Aspects of Axiomatics," presented at 42nd General Assembly of CIRP, Aix-en-Provence, France, 1992.

[Taylor 1911] F. W. Taylor, "The Principles of Scientific Management," 1911, URL: http://melbecon.unimelb.edu.au/het/taylor/sciman.htm.

[Thomas 2004] D. Thomas, "Furniture Styles and Periods in History," 2004, URL: http://www.furniturestyles.net/.

[Tichkiewitch 1990] S. Tichkiewitch, "Quelques problèms posés par la conception intégrée," in *Calcul Des Structures Et Intelligence Artificielle*, vol. 4, J. M. Fouet, L. P., and R. Ohayon, Eds. Paris: Pluralis, 1990, pp. 251-262.

[Tichkiewitch 1994] S. Tichkiewitch, "De la CFAO à la conception intégrée," *International Journal of CAD/CAM and Computer Graphics*, vol. 9, pp. 609-621, 1994.

[Tichkiewitch 1996] S. Tichkiewitch, "Specifications on integrated design methodology using a multi-view product model," presented at Engineering Systems Design and Analysis Conference, Montpellier, 1996.

[Tichkiewitch et al 2006] S. Tichkiewitch, B. Radulescu, G. Dragoï, and K. Pimapunsri, "Knowledge Management for a Cooperative Design System," in *Advances in Design, Springer Series in Advanced Manufacturing*, H. A. ElMaraghy and W. H. ElMaraghy, Eds. London: Springer London, 2006, pp. 97-107.

[Tollenaere et al 1995] M. Tollenaere, P. Belloy, and S. Tichkiewitch, "A part description model for the preliminary design," in *Advanced Cad/Cam Systems: State-Of-The-Art and Future Trends in Feature Technology*, R. Soenen and G. J. Olling, Eds.: Kluwer Academic Pub, 1995, pp. 129-143.

[U.S. Census Bureau 2006] U.S. Census Bureau, "World Population Information," 2006, URL: http://www.census.gov/ipc/www/world.html.

[UEA 2005] UEA, "The World Furniture Industry: Production-Trade-Market," UEA: European Furniture Manufacturers Federation, 2005, URL: http://www.tmf.se/litiuminformation/site/page.asp?Page=64&IncPage=4865&Destination=4815.

[UK Technology Education Centre 1996], "What is Design?," 1996, URL: http://atschool.eduweb.co.uk/trinity/watdes.html.

[UNECE 2005] UNECE, "European Forest Sector Outlook Study - Main Report," UNECE: United Nations Economic Commission for Europe, Geneva, Switzerland AE428/E, 2005.

[WhatIs.com 2005] WhatIs.com, "Definition of taxonomy," 2005, URL: http://searchsmb.techtarget.com/sDefinition/0,,sid44_gci331416,00.html.

[Wikipedia 2007] Wikipedia, "Particle board," 2007, URL: http://en.wikipedia.org/wiki/Particle_board.

[Willemse 1997] M. A. Willemse, "Interactive product design tools for automated assembly." Delft, Netherlands: Delft University of Technology, 1997, pp. 170, ISBN: 90-370-0162-9.

[Windchill] PTC, URL: http://www.ptc.com/appserver/mkt/products/home.jsp?k=37

[Yang and Zhang 2000a] K. Yang and H. Zhang, "A Comparison of TRIZ and Axiomatic Design," *The TRIZ Journal*, 2000, URL: http://www.triz-journal.com/archives/2000/08/d/index.htm.

[Yang and Zhang 2000b] K. Yang and H. Zhang, "Compatiability Analysis and Case Studies of Axiomatic Design and TRIZ," *The TRIZ Journal*, 2000, URL: http://www.triz-journal.com/archives/2000/09/c/index.htm.

[Youngquist 1996] J. A. Youngquist, A. M. Krzysik, P. Chow, and M. Roger, "Properties of composite panels," in *Paper and Composites from Agro-Based Resources*, R. M. Rowell, R. A. Young, and J. K. Rowell, Eds.: CRC Lewis Publishers, 1996, pp. 331-336, ISBN: 0-160-00043-2.

[Youngquist 1999] J. A. Youngquist, "Wood-based Composites and Panel Products," in *Wood handbook—Wood as an engineering material*, F. P. Laboratory, Ed., Rev. 1999 ed. Madison, Wisconsin: Madison, WI: U.S. Department of Agriculture, Forest Service, Forest Products Laboratory, 1999, pp. 10-1—10-31, ISBN: 1-566-70235-6.

Résumé

1. Introduction

Pouvons-nous éviter de la « globalisation » en parlant du développement du processus de conception? Quelle est la vraie signification du terme « globalisation » dans le cadre du processus de conception? Pourquoi devons-nous être concernés par la globalisation? Il est légitime de se poser toutes ces questions. En outre, «ingénierie concourante», «collaboration», «intégration» peuvent être inclus dans le sujet. Rappelons le but de la conception de produits dans laquelle se posent les questions fondamentales dont nous devons tenir compte: qualité, coût, temps, et recyclage (QCTR). Dans les années 70, le coût du produit était le levier principal pour l'avantage concurrentiel. Le producteur désire réduire le coût d'un produit (coût du matériel, coût de la main-d'œuvre, coût de transport, etc.). Beaucoup de producteurs ont établi leurs usines là où les ressources étaient meilleur marché et faciles à acquérir. Nous pouvons considérer qu'avec ce mouvement a commencé la globalisation. Plus tard dans les années 80, le coût a été concurrencé par la qualité qui est devenu une question primordiale. Des techniques et méthodes ont été conçues et développées pour améliorer la qualité du produit. Plus tard, en raison de raccourcissement du cycle de vie du produit et de la concurrence intense, les exigences des clients augmentent. Le client ne se concentre plus seulement sur le faible coût unitaire et sur la qualité des produits mais des facteurs tels que le délai sur la mise en marché ou les questions de personnalisation de produit prennent un rôle principal et définissent le succès des entreprises. Aujourd'hui, le recyclage est devenu une des questions principales. Il concerne tous les deux problèmes de sauvegarder de l'environnement et des économies de matière première.

La globalisation devient de plus en plus prédominante puisque la technologie a été élargie, en particulier en informatique et communication. L'environnement global du marché et de la fabrication a été mentionné concernant les facteurs d'économie et les avantages concurrentiels. En conséquence, beaucoup d'usines ont été décentralisées vers des pays à bas coût de main-d'œuvre et bas coût matériel, afin de réduire le prix unitaire de produit. Ainsi, certains membres de l'équipe de conception doivent être localisés en différents endroits. En outre, pour raccourcir le temps de la conception à la production, les phases de développement de produit doivent être exécutées en recouvrement. Ainsi, «l'ingénierie concourante», «la collaboration », et «l'intégration » sont de plus en plus mentionnées.

L'approche de l'ingénierie concourante vise à réduire le délai de la mise sur le marché du produit, à développer le processus de production et à réduire les coûts, en exécutant diverses activités technologiques en parallèle avec une équipe multifonctionnelle. Néanmoins, en raison de l'augmentation de la complexité de la conception des produits, de l'approche CE[1]

[1] CE - Concurrent Engineering = l'ingénierie concourante

découlent de nombreuses décisions qui peuvent mener des conflits entre les acteurs plus tard. Dans cette situation, l'approche de l’ingénierie collaborative est devenue nécessaire. Cette approche vise à soutenir les individus de l'équipe de conception pour travailler ensemble vers un but commun et à trouver des solutions qui satisfassent tout le monde. Cette approche facilite le travail des acteurs en les introduisant dans un environnement de collaboration et en leur donnant des moyens de communication pour résoudre les conflits de la conception. Cependant, elle n'assure pas que les acteurs puissent réaliser efficacement les activités de la conception. En outre, quelques malentendus au cours des réunions peuvent mener à augmenter le temps du développement et le coût. Actuellement, on propose un pas supplémentaire avec la conception intégrée. Dans le contexte de la conception intégrée, n'importe quel acteur qui doit intervenir à un moment quelconque du cycle de vie de produit se doit d'être présent dans le processus de conception afin de partager et d'échanger ses informations avec l'équipe pour développer la conception du produit. La conception intégrée permet de fusionner les compétences d'acteurs différents, favorisant la résolution des contradictions entre les disciplines, et puis d'intégrer la connaissance dans la conception du produit.

L'industrie du meuble en bois est l'un des secteurs les plus avancés parmi ceux qui ont effectué la globalisation. Ses produits ont un cycle de vie court et changent rapidement de modèle. Dans nos études, nous proposons une conception intégrée de meubles en bois réalisés en panneau de fibres ou de particules. La croissance en terme de marché de ces meubles est en augmentation régulière. Les facteurs principaux de cette croissance sont le bas prix du produit et la proposition de prêt-à-assembler (meuble en kit). Par contre, ces facteurs introduisent les entreprises concernées dans l’environnement plus complexe.

Dans le processus de conception, les acteurs peuvent rencontrer des difficultés en recueillant des informations, en communication, en coopération, et/ou en prenant des décisions en raison de la délocalisation. Ceci pourrait entraîner des processus de reconception et retarder le délai pour la mise sur le marché. En outre, les compagnies ont besoin d'être plus concurrentielles et mettent un effort supplémentaire dans le système pour satisfaire aux exigences du client autant que faire se peut, même si elles créent par la même une complexité additionnelle. Par conséquent, le processus de conception doit être développé pour résoudre la complexité de la conception des produits.

Nous soulignons que l'équipe de conception a besoin d'un espace qui permette aux acteurs de communiquer, de partager et d'échanger des informations pour résoudre les problèmes et la complexité de conception. Nous proposons pour cela dans nos études un modeleur coopératif de conception (CoDeMo). Un des objectifs principaux de CoDeMo est de créer un environnement collaboratif comme une salle de réunion virtuelle qui permette aux différents acteurs, connectés au réseau, de participer à un projet de conception en synchrone ou asynchrone. Grace aux méthodes et aux modèles de l'intégration, le système permet aux acteurs d'utiliser leurs connaissances dans le projet de conception, d’accéder à une base de données commune, d’échanger leurs informations, de discuter sur des problèmes de conception, de négocier et de réaliser des compromis pour résoudre la complexité de

conception. Chaque acteur peut également utiliser ses applications spécifiques pour résoudre des problèmes de conception et/ou pour évaluer la conception.

Cette thèse se compose de trois parties.

La première partie correspond à l'état de l'art sur les meubles réalisés en panneaux de fibres ou de particules, les philosophies de l'ingénierie de conception, et le principe de l'« *Axiomatic Design* ». Dans un premier temps, elle présente l'idée générale, des observations sur la croissance du marché de meubles en bois et l'importance d'étudier la conception de ce type de meuble. En suite, elle examine les approches existantes et courantes des processus de conception. Elle précise la problématique et les difficultés du processus de conception, ainsi que les limitations des approches. A la fin de la première partie est présenté le principe de l'*Axiomatic Design*. Cette dernière décrit intensivement la problématique de conception en introduisant la théorie de la complexité de Nam Suh. La complexité inhérente au processus de conception est par conséquent examinée.

La deuxième partie présente les concepts de la conception intégrée. Elle présente les modèles et les méthodes pour l'intégration qui ont été développées dans cette étude et par l'équipe de conception intégrée du laboratoire G-SCOP. Ces méthodes et modèles soutiennent le système pour créer un environnement collaboratif et permettent donc aux acteurs de disciplines différentes de travailler en collaboration. Nous appliquons le concept du modèle produit pour stocker les données du produit et la connaissance liant les compétences différentes. Nous appliquons également le concept de multidisciplinarité pour faciliter aux acteurs la présentation de leurs informations et contraintes pendant le processus de conception. En fait, l'équipe de conception se compose des différents acteurs venant de différents domaines de compétence. Cette partie présente la constitution d'un modèle de connaissance utilisé par les acteurs et présente comment ces acteurs contribuent à créer de l'information dans le processus de conception. En outre, nous proposons une méthode permettant de résoudre la conception de problèmes imaginaires complexes et indépendants du temps

La troisième partie vise à valider le système de conception intégrée et ses applications spécifiques en présentant une application pour des meubles réalisés en panneaux de fibres ou de particules. Elle valide l'utilisation des *entités*[2] et des *règles de production* et présente des interactions entre les acteurs de différentes vues pendant le processus de conception. Dans cette étude, nous tenons compte principalement de trois domaines de compétence : l'assemblage, la mécanique, et l'usinage. Par conséquent l'interaction entre les acteurs de ces domaines est présentée. Elle présente également comment les acteurs traitent des problèmes de conception dans le contexte de la conception intégrée.

La conclusion résume les résultats de cette étude. Elle présente également des projections des travaux futurs devant être développés.

[2] Le mot « entité » peut être se traduire en anglais « entity » mais dans ce cas il s'agit de « feature ».

2. Etat de l’art

2.1 A propos des meubles réalisés en panneaux de fibres ou de particules

L'industrie du meuble est l'une des industries les plus importantes dans l'environnement de production global. En 2005, la production mondiale des meubles pèse environ 220 milliards d'euros. On prévoit que la croissance du secteur l'amènera à 1000 milliards d'euros en 2050 [De Turck 2005]. Pendant la période de 1995-2005, les Etats-Unis, plus grand importateur de meubles du marché unique au monde, accroissaient son marché de 6.5 milliards de $ à 23.8 milliards de $ [FFE 2006]. Plus de 60 % de tous les meubles ménagé importés sont en bois. Au séminaire d’"Outlook for the Furniture Markets" organisée par CSIL Milano en Italie, la prévision du commerce international des meubles a prévu d'atteindre 82 milliards de $ en 2005, puis 90 milliards de $ en 2006 et 97 milliards de $ en 2007.

L'Europe était le plus grand marché de meubles au monde. En 2004, l'Europe basée sur un marché de 25 pays, a une consommation apparente totale de meubles de 95.6 milliards €, en augmentation de 1.1% par rapport à l’année 2003. Cette production représente 43.1% de la production globale [UEA 2005]. Les observations des futurs marchés indiquent que la tendance de croissance des meubles en bois dans la future décennie augmentera continuellement.

En se concentrant sur les tendances et les projections pour la production du panneau de particules et du panneau de fibres agglomérées en Europe, [UNECE 2005] prévoit que la production de panneau de particules augmentera à un taux annuel moyen de 2.6 %, avec une augmentation de production de 40 millions m^3 en 2000 à 67 millions m^3 en 2020. La production de panneau de fibres augmente aussi à un taux annuel moyen de 3.1 %. En plus, elle doublera presque au cours des 20 années à venir, de 12.7 millions m^3 en 2000 à 23.5 millions m^3 en 2020.

2.2 Avantages d’utilisation des panneaux de fibres ou de particules

En raison de la raréfaction des arbres, la demande de panneaux de fibres ou de particules est prévue pour augmenter. Ils sont actuellement utilisés couramment dans des applications diverses, souvent de manière parallèle au bois massif. D'ailleurs, ils sont préférés au bois massif dans certaines applications, en raison d'avantages compétitifs :

- Personnalisation pour des applications : puisque des panneaux de fibres ou de particules sont en bois reconstitué, ils peuvent être conçus pour répondre aux exigences spécifiques à l'application telle que dimension, forme, propriétés mécaniques. En plus, en utilisant les matériaux synthétiques pour les placages et bandes de bord afin de recouvrir les panneaux, les producteurs peuvent concevoir leurs produits de manière plus diverse et attrayante.

- Acquisition : en raison des restrictions d’abattage des forêts dans beaucoup de pays, il est devenu de plus en plus difficile d'obtenir du bois massif pour fabriquer des meubles en bois. A contrario, les panneaux n'ont pas besoin d'un grand tronc pour fabriquer une grande surface.

- Coût : les panneaux sont moins chers de par leur fabrication à grande échelle car ils sont faits à partir de particules de bois telles que de la sciure, des petits morceaux de bois, des résidus de bois, etc.

- Conservation de l'environnement : Bien que des panneaux peuvent poser certains problèmes de pollution de par l'utilisation de la résine synthétique ou d'adhésif, ils sont plus environnemental-amicaux pour la forêt que le bois massif en raison de l'utilisation des chutes et résidus de bois.

2.3 Evolutions du processus de conception

Notre étude concerne l'élaboration des méthodes et des outils qui permettent à des acteurs de travailler en collaboration et intégration. Cependant, avant que l'on puisse proposer un tel système, il est nécessaire de réaliser un arrangement de la façon dont on va développer le processus de conception et les méthodologies correspondantes.

Le terme "processus de conception" peut vouloir être considéré comme "processus de résolution des problèmes", ce qui commence par l'identification et l'analyse d'un problème ou d'un besoin. La « conception » n'est pas un processus total, ni une activité pour seulement des ingénieurs et des concepteurs mais c'est une activité partagée entre ceux qui conçoivent des objets, des systèmes et des environnements, ceux qui les développent et les réalisent et ceux qui les emploient.

En parlant de révolution du processus de conception, nous découvrons des approches comme "l'ingénierie séquentielle"," l'ingénierie concurrente", et "l'ingénierie collaborative". La différence de ces approches est développée dans [Lu 2006] et ici décrite dans le figure 1.

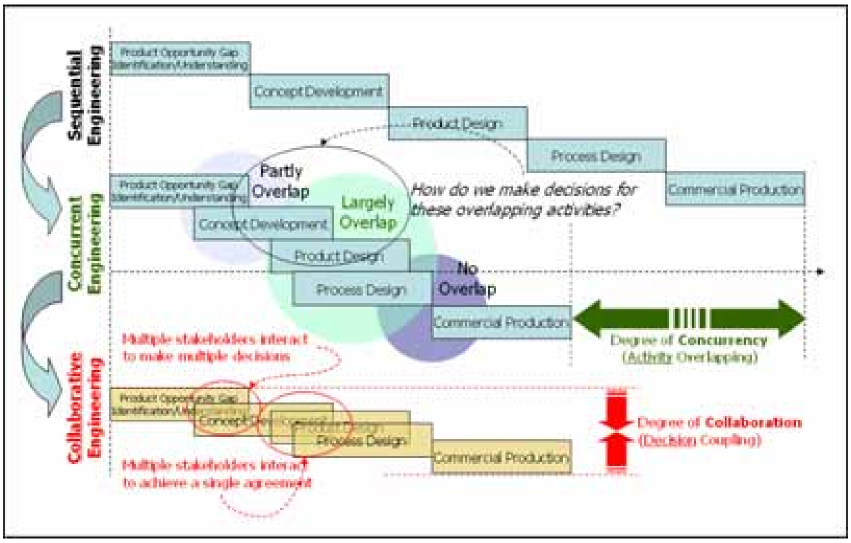


Figure 1 Développement de processus de conception [Lu 2006]

L'ingénierie séquentielle a été remplacée par l'approche CE pour pouvoir développer une meilleure qualité, atteindre un coût inférieur et réaliser une conception plus rapide. L'approche CE résout le problème du "over-the-wall" [Salomone 1995] par l'exécution de phases différentes en parallèle avec des équipes multifonctionnelles. Elle réalise modérément le délai d'exécution de la phase conception en augmentant le degré de simultanéité entre les phases. Cependant, la complexité des produits ayant augmenté, le nombre de décisions a augmenté tout autant si ce n'est plus. Dans cette situation, une approche de collaboration est devenue nécessaire. L'ingénierie collaborative diminue l'espace entre les phases de conception, en augmentant le degré de collaboration parmi des individus et des équipes, en incluant des perspectives de négociation et en faisant des compromis pour avoir concordance de vue. Le processus de conception engage des disciplines diverses: marketing, technologue, assemblage, mécanique, usinage, maintenance, recyclage, etc., chacune ayant un expert dans son domaine concerné par un objectif différent. En introduisant les acteurs dans un environnement collaboratif, on assure pas pour autant qu'ils puissent collaborer et exécuter efficacement les activités de conception. Sky et Buchal [Sky et Buchal 1999] ont identifié que les réunions sont la méthode principale pour résoudre des contradictions et des conflits, et donc des malentendus au cours de ces réunions peuvent mener aux augmentations de temps de développement et des coûts de conception. Donc, le système doit soutenir les acteurs pour intégrer la connaissance de différentes disciplines. En conséquence, nous proposons une approche de conception intégrée dans cette étude.

Celle-ci vise à développer un processus de conception intégrée en prenant les avantages de l'approche de collaboration de technologie. Tichkiewitch a présenté dans [Tichkiewitch 1990], une nouvelle vision de la conception qui inclut des modèles venant de domaines différents, un modèle produit "intelligent" et des gestionnaires de base de données qui tiennent compte d'une multitude d'experts. [Tichkiewitch 1994] présente également le développement du processus de conception en démarrant à la période de la CFAO et aboutissant à une approche de conception intégrée. Cette approche tient compte entre autre de la phase de fabrication pendant le processus de conception (et toutes les phases de la vie du produit) afin d'optimiser le produit final. Un objectif principal de la conception intégrée est de réduire des itérations de conception en tenant compte des contraintes des différentes disciplines dès que possible, et avant de prendre une décision. Il signifie que des contraintes contradictoires peuvent être identifiées et résolues au plus tôt, contrairement à une approche de conception non-intégrée [Roucoules et al 2003].

Dans [Sohlenius 1992] sont définis les trois facteurs les plus critiques : complexité, qualité, et délai d'obtention, facteurs qui déterminent la compétitivité d'un développement de produit, tel que représenté figure 2. Sohlenius dit également que pour rester concurrentiel, un produit doit intégrer des fonctions multiples avec succès afin de traiter et de réduire la complexité et de toujours répondre à des exigences fonctionnelles.

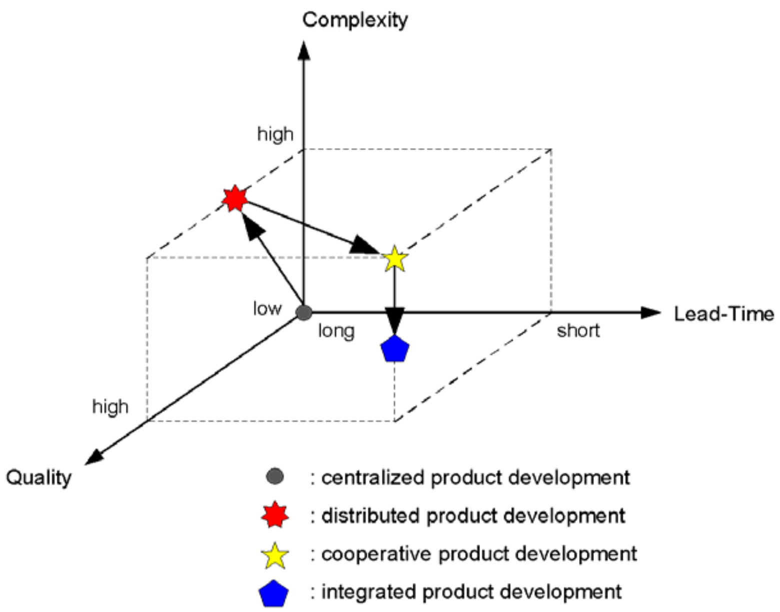


Figure 2 Révolution de développement de processus de conception[3]

2.4 La complexité en conception

Par suite de la globalisation, les problèmes d'ingénierie deviennent aujourd'hui de plus en plus complexe. La complexité dépend de la capacité à synthétiser les problèmes. Pour réduire la complexité, un des axes est de remplacer l'approche empirique par une approche plus scientifique. Suh propose une théorisation de la conception dans ses livres "the principles of design" [Suh 1990], "Axiomatic Design" [Suh 2001], et "A Theory of Complexity and Applications" [Suh 2003]. Il présente la conception comme un processus de transformation entre quatre domaines : le domaine du client, le domaine fonctionnel, le domaine physique, et le domaine du processus de fabrication, tels que présentés dans la figure 3.

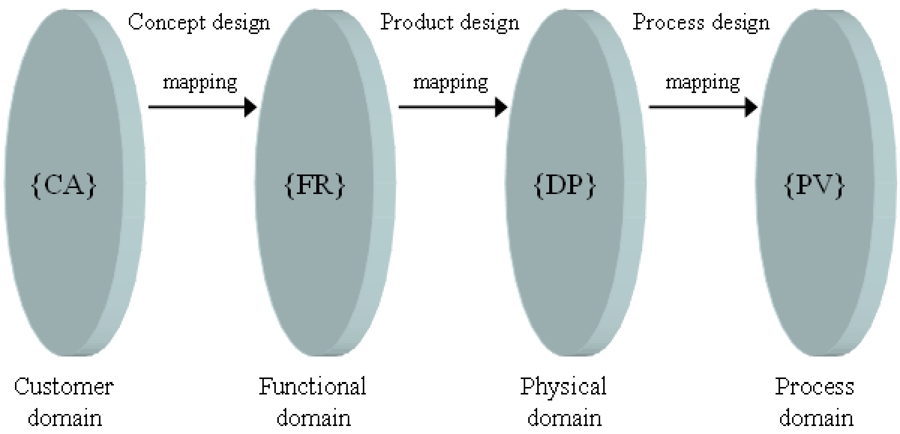


Figure 3 Quatre domaines du monde de conception [Suh 2001]

[3] Cette figure est modifiée de [Sohlenius 1992]

Il s'appuie ensuite sur deux axiomes qui permettent de bien concevoir :

Axiome d'indépendance : maintenir l'indépendance des fonctions (FRs)[4].

Une conception optimale doit, à tout moment, maintenir l'indépendance des fonctions. Dans une conception acceptable, les paramètres de conception (DPs)[5] et les FRs sont reliés de façon que l'on puisse ajuster un paramètre de conception spécifique (DP) pour qu'il satisfasse à la fonction correspondante sans affecter les autres FRs.

Axiome d'information : minimiser le contenu d'information de la conception.

La meilleure conception est une conception fonctionnellement non couplée, celle dont le contenu d'information est minimum.

Dans l'axiomatique de Suh, un formalisme mathématique est construit en termes de vecteurs caractéristiques. Un group de FRs constitue un vecteur {FR} tandis qu'un group de DPs dans le domaine physique constitue un autre vecteur {DP}. La relation fonctionnelle entre ces deux vecteurs est alors donnée par une équation: $\{\mathbf{FR}\} = [A]\{\mathbf{DP}\}$, où [A] est un groupe de caractéristiques de *la conception de produits* qui s'appelle "matrice de conception". Pour la conception de processus, un group de PVs constitue un vecteur {PV}. La relation fonctionnelle entre le domaine physique et le domaine du processus de fabrication est alors donnée par une équation : $\{\mathbf{DP}\} = [B]\{\mathbf{PV}\}$, où [B] est un groupe de caractéristiques de *la conception de processus*.

Selon l'*Axiomatic Design*, la complexité est liée à l'information: plus est complexe un produit ou un système, plus l'information sera importante. [Suh 2001] a classifié la complexité dans deux catégories : complexité dépendante du temps et complexité indépendante du temps. La complexité indépendante du temps est divisée en vraie complexité et complexité imaginaire tandis que, la complexité dépendante du temps est divisée en deux types différents : complexité combinatoire dépendante du temps et complexité périodique dépendante du temps. Notre étude s'intéresse particulièrement à la complexité imaginaire indépendante du temps. Ce type de complexité arrive souvent quand nous devons satisfaire plusieurs FRs dans le processus de conception. Pour résoudre la complexité imaginaire, nous présentons une méthode dans la session suivant.

3. Vers la conception intégrée

3.1 Modèles de la conception par intégration

Le développement explosif de l'Internet et des technologies d'information et de communication a introduit un problème de surcharge de l'information. Il est toujours plus difficile à faire face à toute nouvelle information qu'on reçoit [Heylighen 2002]. Comment contrôlons nous l'information incluant aussi bien les pertinentes et les non pertinentes? Suite à

[4] Functional Requirements

[5] Design Parameters

l'étude de Chapa Kasusky [Chapa Kasusky 1997], nous appliquons le concept du *modèle de produit* sur ce travail pour gérer les informations pendant le processus de conception. Le modèle de produit, dans le contexte de l'intégration, est un modèle informatique qui se compose d'un modèle de connaissance et d'un modèle de données. Il intègre alors des entités, éléments du modèle factuel de connaissance, dans une structure respectant les règles et la grammaire du modèle de données. La structure du modèle de produit associé par le modèle de données et le modèle de connaissance est représentée dans la figure 4.

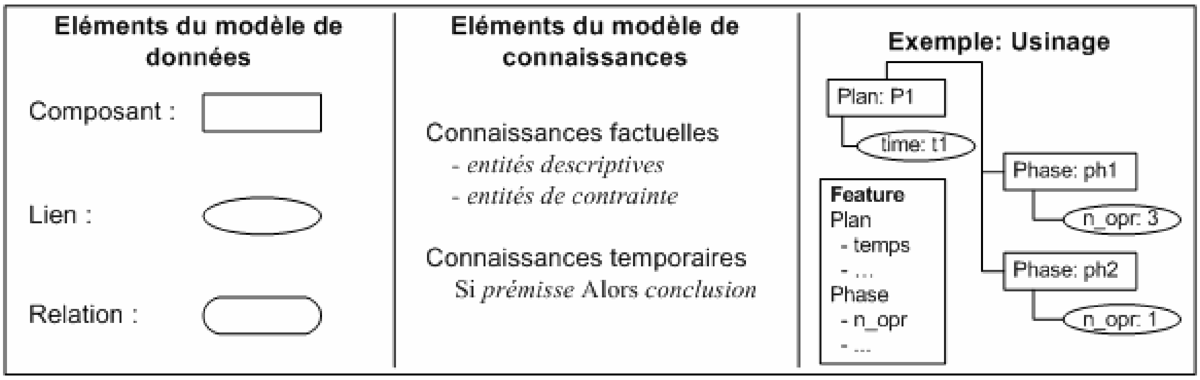


Figure 4 La structure du modèle de produit

3.1.1 Le modèle de données

Le modèle produit est basé sur une structure de données et sur des opérateurs que l'on peut y appliquer. Le modèle de données que nous proposons est un modèle multi-vue. Nous rappellerons ici rapidement les trois éléments de base de ce modèle de données : le composant, le lien et la relation.

Le *composant* est un élément granulaire indispensable pour la description d'un produit. Il peut être un composant physique, élément matériel du produit (carter, arbre d'une boîte de vitesses, roulement,…), ou un composant essentiel à la modélisation du produit (maillage en élément finis pour un calcul de comportement,…). Il peut également être un élément temporaire dans le cycle de vie du produit (forme brute avant usinage,…).

Le *lien* est une partie du composant permettant à son environnement de le percevoir. Le lien n'existe pas sans son composant. Plusieurs liens du même composant peuvent être indépendants ou en recouvrement.

Un composant, pour pouvoir être perçu de l'extérieur, a donc besoin d'avoir des liens. Les liens formalisent ainsi l'interface du composant avec son environnement, ce dernier pouvant appartenir à la même vue que le composant que le lien caractérise, ou à une vue différente. L'association entre deux composants se fait alors obligatoirement à travers des liens par ce qu'on appelle une *relation*. Une relation permet l'association de deux ou plusieurs liens, pouvant appartenir à des composants différents ou au même composant. Pour représenter ces éléments graphiquement, nous adopterons le graphisme de la figure 5.

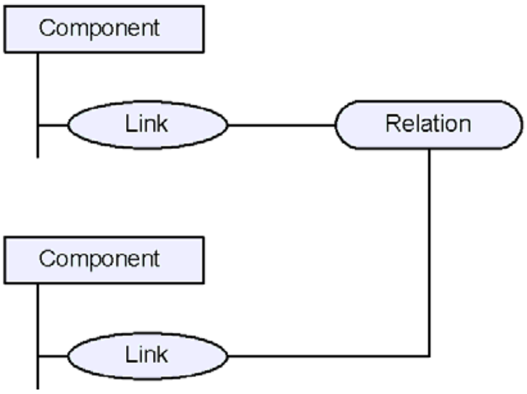


Figure 5: Symbolique graphique et formalisation du modèle de données

3.1.2 Le modèle de connaissances

Chaque acteur de l'activité de conception doit utiliser ses propres connaissances pour pouvoir exercer son métier. Nous proposons alors un modèle de connaissances composé de deux formes : les connaissances factuelles et les connaissances temporaires.

1) Les connaissances factuelles

Les connaissances factuelles sont des connaissances relatives aux données, aux faits pouvant être modélisées par des entités. Nous définirons par "entité", l'objet sémantique qui est manipulé par des acteurs pour décrire un élément servant à définir l'objet conçu. Une entité a un nom, donné par son utilisateur, ce qui la rend dépendante du contexte de sa création. Une entité est décrite par un certain nombre de caractéristiques et de savoir-faire. Nous pouvons plus loin diviser des entités dans deux catégories suivantes:

Les entités descriptives décrivent le produit avec le vocabulaire spécifique selon le point de vue d'un métier spécifique. Par exemple, l'entité de *cylindre* décrit une forme cylindrique du produit avec ses caractéristiques, par exemple: rayon, longueur, et aire. Un comportement de cette entité *cylindre* pourrait impliquer la valeur de l'aire à la valeur du rayon et de la longueur. Noter qu'une entité descriptive peut représenter ou non une forme matérielle. Par exemple, des dispositifs géométriques de forme tels que le cylindre, le rectangle, le cercle, etc., sont des entités matérielles utilisées dans la vue géométrique tandis que des entités de fabrication tels que la coupe, la forge, le fraisage, etc., sont des entités non matérielles utilisées dans la vue de fabrication.

Les entités de contrainte définissent une contrainte pour des caractéristiques d'entités descriptives. Par exemple, l'égalité est une entité de contrainte qui est défini avec deux caractéristiques : variable1 et variable2. Son comportement impose que ces deux caractéristiques doivent être égales. De plus, [Gaucheron 2000] classifie des caractéristiques d'entités par proposer des taxonomies en trois catégories :

- Vernaculaire: se dit d'une entité spécifique à un métier unique : c'est le cas par exemple de la « pression de fermeture des matrices » qui est propre au métier de la forge.

- Véhiculaire: se dit d'une entité partagée entre plusieurs métiers : c’est le cas de l’entité « cordon de bavure » qui est partagée entre les métiers forge et usinage.
- Universelle: se dit pour une entité véhiculaire dans la mesure où elle est compréhensible, partagée, par tous les métiers : c’est le cas de l’entité cylindre.

2) Les connaissances temporaires

Les connaissances temporaires sont des connaissances relatives au traitement du problème, à l’activité de conception, pouvant être modélisées par des règles de production. Une règle de production est un élément du modèle d’activité. Formée d’une prémisse (Si A) et d’une conclusion (Alors B), une règle de production permet de faire évoluer un problème en signalant qu’il faut effectuer les instructions B lorsque les faits A sont avérés. La règle de production crée donc une notion d’évolution, de temporalité, dans la résolution d’un problème.

L’élément déclencheur A peut correspondre au fait que l'instance d’une entité particulière existe, ou que la valeur d’un (ou de plusieurs) de ses attributs réponde à une condition spécifique. A ne peut donc être vrai en absolu, mais dépend obligatoirement d'un contexte donné. Les instructions B peuvent fixer une valeur pour un attribut d'une instance d'une entité particulière, ou prendre en compte une nouvelle entité comme élément du produit, ou déclencher un programme de calcul spécifique qui lui-même donnera lieu à des mises en valeurs ou à des créations.

Pour utiliser le jeu de règles de production correspondant à son métier, chaque acteur devra avoir à sa disposition un moteur d'inférence lui donnant sur demande les règles pouvant être validées. Il pourra alors choisir le fonctionnement de son système en faisant dérouler les règles suivant un mode de contrôle à fixer (marche avant, arrière ou mixte), ou en choisissant un mode manuel pour déclencher telle ou telle règle.

3.2 Méthodes de la conception intégration

Dans le processus de conception, idéalement, toutes les informations et contraintes pertinentes des acteurs doivent être rassemblées avant que l'on puisse prendre une décision. Cependant, avec la globalisation, des acteurs ont été décentralisés dans différents endroits. Ceci amène des difficultés de communication, de partage/échange d'informations. En plus, l'acquisition de l'information pertinente, cohérente et mise à jour au sein d’une grande entreprise prend du temps et est complexe. Nous présentons alors des méthodes qui soutiennent des acteurs pour organiser des activités de conception, et également pour intégrer les connaissances des disciplines différentes au processus de conception.

3.2.1 Une méthode de conception intégrée

Dans le contexte de la méthodologie de conception intégrée, le processus de conception peut être divisé en deux phases. La première phase de conception dans laquelle les technologues font appel à leurs connaissances afin de réaliser les choix technologiques du système. Ces choix doivent répondre au besoin fonctionnel, c’est-à-dire aux fonctions principales qui sont l'essence même du produit. Cette première phase de conception amène progressivement à la définition des surfaces fonctionnelles d'usage du produit. Ces surfaces sont issues de la technologie choisie, et représentent les surfaces à travers lesquelles circulent les flux énergétiques pour réaliser les fonctions principales. A partir de ces surfaces fonctionnelles d'usage, les autres métiers de la conception apportent leurs expertises et leurs contraintes pour définir le produit. Ils réalisent, en deuxième phase de conception, la description complète du produit en intégrant leur propre point de vue et dimensionnement pas à pas le projet.

La conception intégrée a pour objectif de prendre en compte tous les métiers ayant, à un moment ou à un autre, à intervenir dans le cycle de vie du produit. Il est évident que la première phase de conception amorce le processus de conception. La deuxième phase de conception est mise en route dès qu'un acteur a suffisamment de données pour pouvoir réagir et apporter sa propre contribution, en juste besoin. Il n'y a donc pas attente de la fin de la première phase de conception pour passer à la seconde mais recouvrement des deux phases.

3.2.2 Les mondes de la conception

Mer propose dans [Mer 1998] le concept de monde dont la définition est la suivante : « Un monde de la conception est un *ensemble hétérogène* regroupant des entités (qui peuvent être des outils, des objets, des personnes) qui développent la même ***logique d'action***, relèvent de la même ***échelle de grandeur*** et partagent des ***connaissances collectives*** ».

La notion de *logique d'action* associe l'enjeu, l'objectif de l'action et l'action elle-même. Elle nous permet de ne pas dissocier le cadre de l'action (les objectifs, les contraintes, sa valeur …) et l'action. De plus, elle signifie qu'il y a continuité entre toutes les actions d'un acteur, que l'on peut y trouver une constante, "un fil conducteur" : une logique. Cependant, elle ne se réduit pas à l'objectif de l'action.

La notion d'*échelle de grandeur* est associée à la notion de logique d'action. Elle permet de légitimer des actions (pourquoi je fais ça), des outils (pourquoi je choisis cet outil), et des objets (pourquoi j'utilise cette opération). Une entité sera d'autant plus grande qu'elle participera et renforcera la logique d'action. Les jugements peuvent porter sur les actions, sur les acteurs, sur les objets ou les outils. Les acteurs jugent le produit, l'évaluent tout au long du processus de développement. De même, ils se jugent les uns les autres à travers leurs actions, passées et présentes. Pour tous ces jugements, ils s'appuient sur l'échelle de grandeur du monde auquel ils appartiennent. Ce n'est pas un "principe de justice" mais un "principe de justesse (d'adéquation de l'action)". C'est dans l'action et le conflit que se repère un monde.

La notion des *connaissances collectives* décrivent les savoirs, les conventions, les règles explicites ou implicites qui sont partagées par tous les acteurs d'un monde. Dans les connaissances collectives se trouve aussi le langage partagé qui permet aux acteurs de se comprendre rapidement. Cette notion regroupe les conventions formelles ou tacites, les représentations du produit et le langage partagé par les entités d'un même monde.

3.2.3 L'objet intermédiaire

[Mer et al 1995] présente le concept de l'objet intermédiaire comme analyseur de l'activité de conception. Les objets intermédiaires ont un *rôle de communication* très important au sein de processus de conception. Non seulement comme support d'information mais aussi, et surtout, comme *instrument de coordination* entre les acteurs. Ces deux aspects sont indissociables.

Il est modélisation de la réalité comme modèle de représentation du futur produit et, en même temps, le processus dont il est le résultat. Cette représentation évolue avec la connaissance croissante relative au projet.

Il est instrument de coordination ou de coopération des acteurs de la conception. Il diminue alors le champ de leurs divergences. Les objets intermédiaires sont au centre des nombreux échanges qui ont lieu durant la conception. Cet aspect nous permet d'introduire différents axes pour caractériser les objets en interaction dans le processus.

L'objet intermédiaire est caractérisé comme un objet commissionnaire ou un objet médiateur. Nous pouvons définir les termes 'commissionnaire' et 'médiateur' à partir des interactions entre le produit et ses utilisateurs: dans sa situation d'action. L'acteur utilisant un objet commissionnaire est en interaction, à travers l'objet, avec les intentions, les idées du producteur de l'objet même si elles sont quelque peu déformées. En revanche l'utilisateur d'un objet médiateur est en interaction avec l'objet lui-même. Dans cette situation, l'objet devient "acteur". Il médiatise, au moins partiellement, le processus de conception antérieur. Représentant une partie de la conception, il fonctionne cependant "par lui-même" et agit comme un acteur à part entière. À la fois, nous définissons une seconde caractéristique de l'objet intermédiaire comme un objet ouvert ou un objet fermé. La notion d'ouverture est liée à un objet laissant à l'utilisateur une marge de manœuvre au sein de laquelle il peut plus ou moins diverger. En revanche, un objet fermé diminue et tend à faire disparaitre cette marge de manœuvre. L'objet ouvert incite à un travail d'interprétation, tandis que l'objet fermé transmet une prescription. Noter qu'afin de permettre l'intégration des différents points de vue métiers, liés à la vie du produit, l'objet doit être le plus ouvert possible.

3.2.4 Le concept de multi-acteurs

Dans le processus de conception, "acteur" ne veut pas que dire le concepteur mais tous les membres de l'équipe participent à la définition du produit. Ils introduisent dans la définition du produit des contraintes imposées par les règles propres au métier qu'ils représentent. Ces intervenants sont les acteurs de la conception. Les différents acteurs intervenant lors de la conception restent à l'heure actuelle les acteurs de la conception linéaire. La conception multi-acteurs prend en compte toutes les phases de la vie du produit. Ceci permet d'améliorer la qualité des produits conçus, de diminuer les délais de conception et les coûts de production.

En outre, dans le contexte du modeleur de conception intégrée, nous caractérisons des acteurs en deux catégories : l'acteur externe et l'acteur interne. L'acteur externe est un acteur humain ou un utilisateur qui participe à la définition du produit pendant le processus de conception. Chaque acteur externe construit une représentation du produit en utilisant les entités propres à son métier. Il formera ainsi la vue du métier correspondant. La définition complète du produit se constitue de ces vues métiers et des vues communes.

L'acteur interne est un acteur informatique. Il exécute des tâches de gestion des vues. Pendant le processus de conception, les acteurs externes doivent contribuer et partager un nombre d'information pour caractériser le produit en utilisant leurs entités. Chaque définition d'entité inclut une référence implicite et des compléments du détail spécifiques. L'acteur interne est alors développé pour associer des entités initiales aux entités correspondantes. Les tâches de l'acteur interne sont de garder la cohérence entre des contraintes ou d'exécuter des tâches propres au système, par exemple la propagation de données, la traduction de données, la propagation de contrainte [Roucoules 1999], la substitution [Radulescu 2005], etc.

3.2.5 Le concept de multi-vues et multi-représentations

Nous ne prenons en compte que les acteurs concernés pas un même objectif. Pour réaliser le concept de multi-acteurs, [Chapa Kasusky 1997] a implémenté le concept de multi-vues qui leurs permet d'apporter des informations pertinentes et de présenter le produit tel qu'ils le voient, chacun dans sa propre vue. Nous pouvons caractériser les vues en deux catégories : les vues "métier" et les vues communes. Une vue métier est utilisée pour représenter l'intérêt d'un métier vis à vis du produit. Elle permet aux acteurs de décrire le produit de manière spécifique par ajout de nouvelles données, modification ou suppression des informations existantes. Alors qu'une vue commune est une vue qui est distribuée systématiquement à tous les acteurs. Nous avons maintenant deux vues communes : la vue ossature et la vue géométrie. La vue ossature stocke des informations relatives aux surfaces fonctionnelles du produit avec des caractéristiques telles que rugosité, tolérances, etc. La vue géométrique stocke des données géométriques, et est finalement les résultats de l'intégration des vues métiers. Ces vues communes utilisent des entités majoritairement de type universelle, alors que les vues métiers utilisent des entités plus véhiculaires, voire vernaculaires.

En plus, pour réaliser les concepts de multi-acteurs et de multi-vues, le concept de multi-représentations est proposé. Il permet aux acteurs de représenter leurs informations et leurs contraintes en donnant leur propre représentation, le modèle de données interne de la figure 5, pouvant être proposé sous forme de représentation graphique 3D dans la vue graphique, de représentation fonctionnelle dans la vue technologique ou de représentation textuelle dans une vue en construction (figure 6).

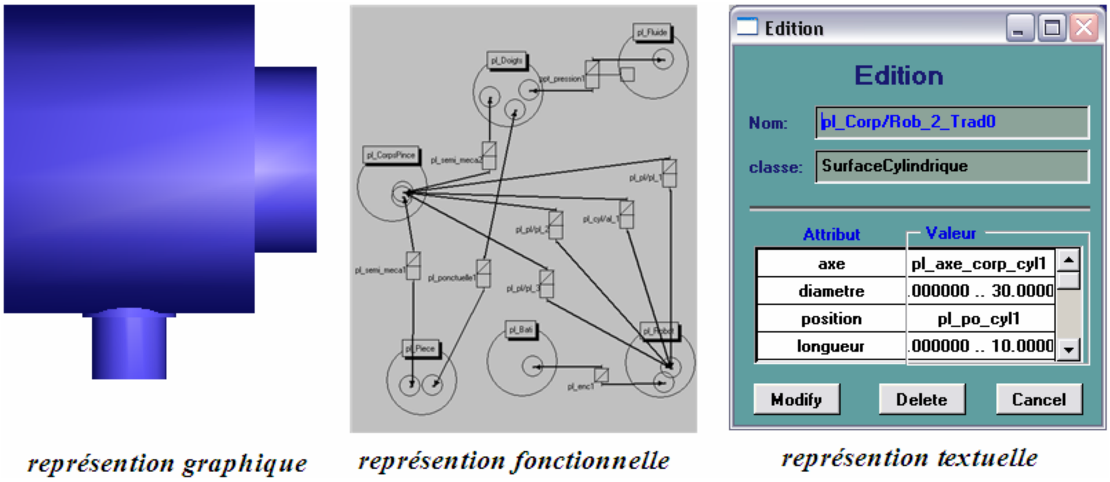


Figure 6 Multi-représentations

3.2.6 Le module de propagation

La complexité imaginaire d'une conception découplée peut apparaître par l'ignorance des concepteurs. Celle-ci peut amener une conception découplée à devenir une conception couplée en raison de l'absent du processus de notification. Pour informer un acteur sur des informations et des contraintes créées par d'autres acteurs, le système doit transmettre celles-ci à la base de données partagée. Dans cette étude, nous appliquons des librairies ILOG pour développer la méthode de propagation. Cette méthode crée une fonction de notification pour notifier des informations créées ou modifiées aux acteurs. Le lecteur peut s'y reporter pour plus d'informations dans [Roucoules and Tichkiewitch 2000].

3.2.7 Le module de traduction

Le modèle produit est constitué de nombreuses informations décidées par des acteurs différents. Nous utilisions des *entités* pour présenter une telle information. Dans le sens vue métier - vue ossature, et vue ossature - vue géométrique, la traduction peut être faite par un module traduction. Pour ceci, deux éléments sont inclus dans le module : une classe "Traduction" et un fichier de données spécifique à cette classe (modifiable par l'utilisateur avec un traitement de texte). La classe, lors de l'instanciation d'une entité, va chercher dans le fichier spécifique si elle-même a une entité associée (ou plusieurs). Si c'est le cas, le traducteur devra instancier l'entité associée (ou les) déclarée dans le fichier. L'avantage de séparer dans un fichier les déclarations du mécanisme de traduction, est de pouvoir déclarer ceux-ci en dynamique au cours de projet.

3.3 Acquisition du modèle de connaissances dans des vues métiers

L'objectif du modèle de connaissance est de permettre aux acteurs de définir les données relatives au produit en cours de conception avec leurs connaissances propres, leur vocabulaire, leurs habitudes. Il s'agit de briques de connaissances spécifiques à chaque métier qui sont totalement indépendantes du produit à concevoir.

Nous présentons ici, par exemple, le cas d'un assemblage d'une planche horizontale à une planche verticale en utilisant une entité '*Cheville*'. Une surface de la planche verticale entre en contact avec un côté de la planche horizontale comme représenté dans la figure 7 (a). Pour utiliser une entité '*Cheville*', nous devons tenir compte de la longueur, du diamètre de la cheville, et aussi de l'épaisseur des deux planches qui doivent être percées pour insérer la cheville, telle que représenté dans la figure 7 (b). En conséquence, nous pouvons créer des connaissances temporelles de l'entité '*Cheville*' telles que celles représentées dans le tableau 1. Cet exemple des connaissances temporelles contiennent des informations cohérences permettant de relier les métiers de l'équipe de conception.

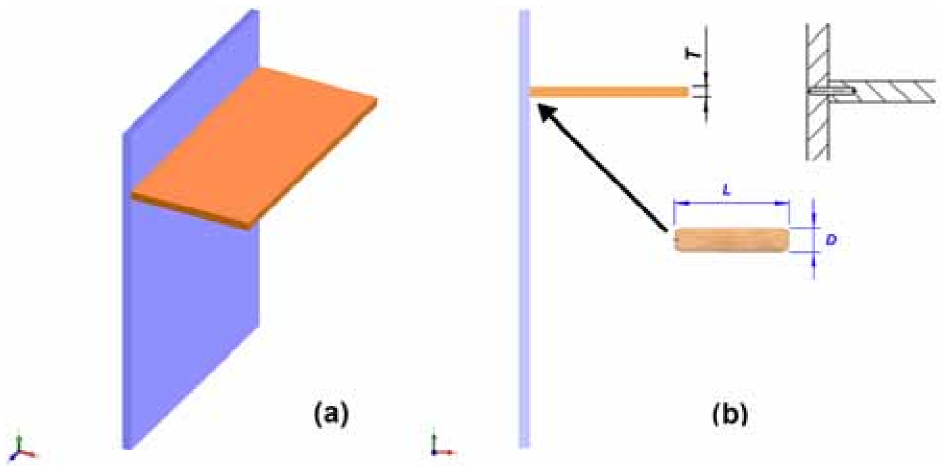


Figure 7 Exemple d'utilisant une cheville en tant que une solution d'assemblage

Tableau 1 Des connaissances temporelles de l'entité Cheville

Si une cheville est appliquée pour fixer une paire de planches
Alors ces deux planches doivent être percées
Si l'épaisseur de la planche horizontale est T millimètres
Alors le diamètre de la cheville n'est pas plus que T/2 millimètres
Si le diamètre de la cheville est D millimètres
Alors ces deux planches doivent être percées avec le diamètre D millimètre
Si la longueur de la cheville est L millimètres
Alors la planche horizontale est percée 2L/3 millimètres tandis que la planche verticale est percée au minimum L/3 millimètres

3. 3.1 Constitution des connaissances dans la vue d'assemblage

Dans la vue assemblage, l'objectif de l'assembleur est d'examiner les possibilités de solution d'assemblage et de choisir la solution la plus acceptable pour assembler les planches. Le système permet de créer une bibliothèque de solutions d'assemblage qui contient les entités spécifiques et leurs caractéristiques comme représentées par exemple dans le tableau 2. L'assembleur doit choisir une solution d'assemblage pour chaque problème d'assemblage détecté dans la vue technologique. Cependant, il ne peut pas définir complètement des valeurs du diamètre d'une cheville tant qu'il n'a pas les épaisseurs des planches correspondantes, qui normalement sont définies par le mécanicien. Cependant, l'assembleur est concerné les propriétés des quincailleries choisies et également par la charge que les planches doivent soutenir.

Tableau 2 Exemples des entités dans la vue assemblage

Entités et caractéristiques		
Cheville Type (Fil, Rainurage) Diamètre Longueur Matériau (Bois, Métal) Maximum de la charge		
Rainure Type (Traversant, Bouchant) Largeur Epaisseur Longueur		

Afin de créer des règles de productions utilisées dans CoDeMo, un fichier neutre est développé pour soutenir une telle connaissance. Il s'appelle QTrans. Ce fichier est développé pour être associé au processus de traduction. Il stocke des solutions possibles à utiliser dans les vues métiers. Nous présentons ici, par exemple, une partie du fichier QTrans, développée pour être utilisé entre la vue assemblage et la vue usinage (tableau 3). Il implique un ensemble de règles mettant en jeu des entités et leurs caractéristiques.

Tableau 3 Règle de production de l'entité Cheville présentant dans le fichier QTrans

```
Component_Name
    Cheville Assem name
Traduction
    Component Percer Usinage name_1_USI
    Component Percer Usinage name_2_USI
    Link name diametre name_diametre
    Link name longueur name_longueur
    Link name_1_USI diametre name_1_USI_diametre
    Link name_1_USI epaisseur name_1_USI_epaisseur
    Link name_2_USI diametre name_2_USI_diametre
    Link name_2_USI epaisseur name_2_USI_epaisseur
    Relation name_diametre name name_1_USI_diametre name_1_USI relation_name_1
    Relation name_longueur name name_1_USI_epaisseur name_1_USI relation_name_2
    Relation name_diametre name name_2_USI_diametre name_2_USI relation_name_3
    Relation name_longueur name name_2_USI_epaisseur name_2_USI relation_name_4
@
```

3. 3.2 Constitution des connaissances dans la vue de mécanique

Dans la vue mécanique, l'objectif principal est de définir l'épaisseur minimum et le type de matériau des planches permettant de supporter les charges. Le résultat de la déflexion d'une planche dépend par exemple d'une part, de la charge appliquée considérant la norme de référence, d'autre part, du type de matériau et de l'épaisseur de la planche. L'objectif de la vue mécanique est de définir le type de matériaux et l'épaisseur des planches. Les choix dans la vue mécanique sont relatifs aux solutions d'assemblage et aux caractéristiques du processus de fabrication, comme présenté avant dans la figure 7 et dans le tableau 1. Nous montrons alors, par exemple, la constitution une règle de production entre la vue mécanique et la vue d'assemblage dans le tableau 4.

Tableau 4 Règle de production de l'entité Cheville entre la vue mécanique et la vue d'assemblage

```
Component_Attribute
        PlancheMeca Meca name_MECH
        Cheville Assem name_ASM

Traduction
        Attribute name_MECH materiau Char materiau_planche
        Attribute name_MECH epaisseur Float epaisseur_planche
        Attribute name_ASM epaisseur Float epaisseur_planche
        Link name_MECH epaisseur name_MECH_epaisseur
        Link name_ASM epaisseur name_ASM_epaisseur
        Relation name_MECH_epaisseur name_MECH name_ASM_epaisseur name_ASM
        relation_MECH_1
@
```

3. 3.3 Constitution des connaissances dans la vue usinage

Dans la vue usinage, le fabricant recueille des informations apportées par les autres acteurs pour planifier le processus de fabrication. Le fabricant peut définir une gamme des processus de fabrication pour chaque planche. Nous présentons ici, par exemple, un diagramme de processus d'opération (OPC) d'un bureau d'ordinateur appelé *DS100* dans la figure 8. Cet OPC donne une vue d'ensemble du processus d'opérations et d'inspections du produit. Il aide le fabricant pour planifier le processus de fabrication et pour évaluer la conception.

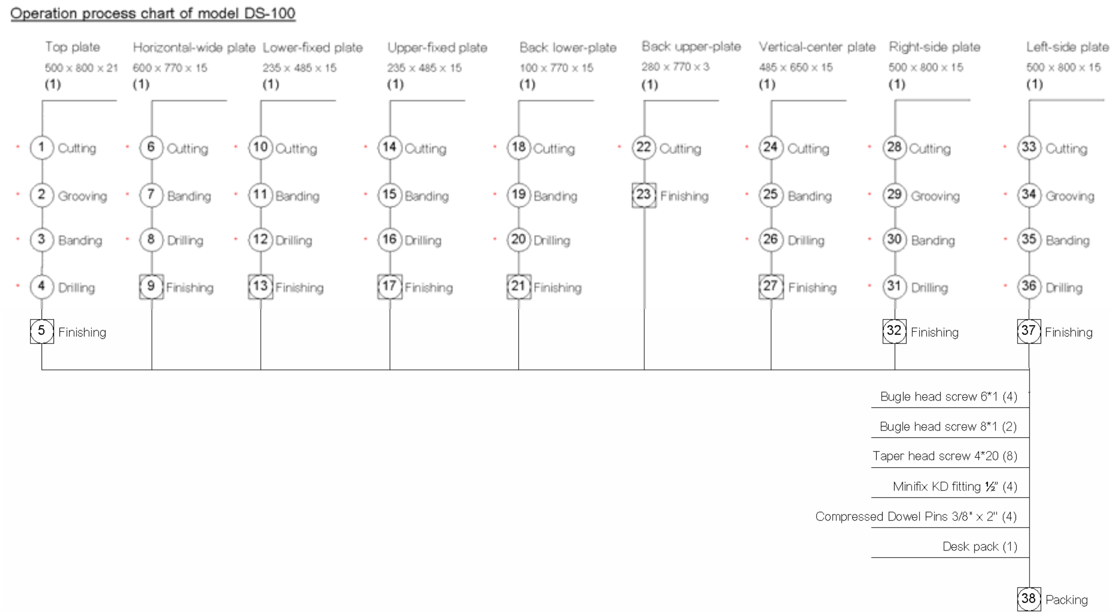


Figure 8 Exemple de processus d'opération

3.4 Intégration pour résoudre la complexité de conception

Afin d'effectuer le processus de conception, les acteurs de l'équipe doivent avoir la notion de coïncidence de la conception, que l'on appelle le « *juste besoin*» [Brissaud et al 1997]:

- Chaque acteur doit apporter ses contraintes dès qu'il pourra. Cette notion permet aux autres acteurs d'avoir plus d'information pour évaluer la conception et pour définir le produit plus avec précision.

- Chaque acteur doit ne doit apporter une contrainte que s'il peut la prouver. Pour souligner la notion précédente, cette notion permet aux acteurs de ne donner seulement une contraintes qu'il peut peuvent justifier celle-ci. Ceci évite d'avoir des choix par hasard.

En outre, nous considérons des hypothèses suivantes dans cette étude:

- Nous supposons que le problème de conception est de complexité imaginaire. En d'autres termes, la matrice inconnue est triangulaire, mais les acteurs ne le savent pas

- Le système de conception intégrée introduit les acteurs autour d'un même environnement. Chaque acteur a sa propre connaissance sur le problème de conception et peut avoir un accès aux données existantes sur le problème.

Nous postulons que si les acteurs travaillent en juste besoin, alors il y a obligatoirement au moins un acteur qui va reconnaître qu'il est capable, tout seul, de résoudre une fonctionnalité (Si les acteurs peuvent résoudre seuls toutes les fonctionnalités, ceci signifie que le problème était en fait découplé, donc non complexe). Cet acteur va donc mettre dans le 'pot commun' la nouvelle donnée qu'il est capable de produire. A partir de là, les autres acteurs prennent connaissances de ce nouveau fait, et l'un d'entre eux doit reconnaitre qu'il est maintenant capable de résoudre seul une nouvelle fonctionnalité.

Par ce principe, on voit que chaque intervention permet de résoudre une fonctionnalité et par la même de réduire la complexité du problème d'un ordre de grandeur. Si le problème est effectivement imaginaire complexe, sans connaître la matrice au départ, la conception intégrée permet la résolution du problème.

Nous montrons ici, par exemple, un développement de conception d'un produit existant (produit de conception), et donc que nous pouvons avoir les matrices de conception. On peut proposer des solutions (DPs) pour satisfaire le FRs. Cette proposition peut être exprimée sous forme d'équation comme:

$$FR_n = aX_1 + bX_2 + \cdots + nX_n \qquad (1)$$

où FR_i est une fonctionnelle et X_i est un paramètre de conception, spécifique à une vue et ainsi à un acteur. Supposons qu'il y a quatre FRs que nous devons satisfaire, comme représentant dans l'équation (2).

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \\ FR_4 \end{Bmatrix} = \begin{bmatrix} X_1 & 0 & X_3 & 0 \\ X_1 & X_2 & 2X_3 & 0 \\ X_1 & 0 & 0 & 0 \\ X_1 & 2X_2 & 3X_3 & X_4 \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \\ DP_4 \end{Bmatrix} \qquad (2)$$

Par la notion de 'juste besoin', les acteurs doivent résoudre ce problème pas à pas, comme représentant dans la figure 9. Ce processus récursif continue également dans le niveau bas de la conception hiérarchique jusqu'à ce que la conception soit complète. En conséquence, il réduit le nombre de FRs insatisfaites à chaque étape et ainsi la complexité dans le processus de conception.

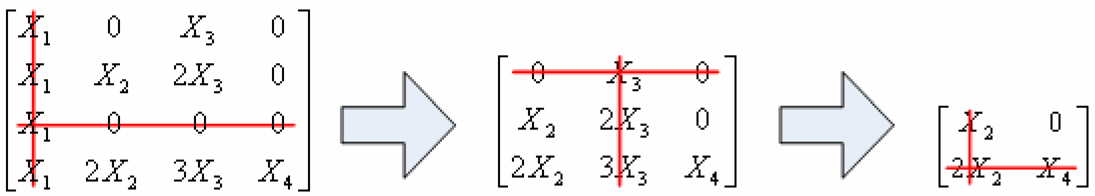


Figure 9 Processus de conception émergente pour résoudre la complexité imaginaire

4. Le système de conception intégrée

Nous appliquons le concept du système de CAID[6] dans cette étude en utilisant CoDeMo pour placer les acteurs dans l'environnement collaboratif. L'architecture du système CAID se compose de trois parties : le système de CoDeMo lui-même, des applications spécifiques, et des interfaces multimédia comme représentés figure 10. Une telle architecture rend le système extensible de façon à pouvoir intégrer un ou plusieurs acteurs externes supplémentaires, augmentant ainsi le niveau global de compétence. Le lecteur peut se reporter pour plus d'informations à [Tichkiewitch 1996] et [Roucoules and Tichkiewitch 2000].

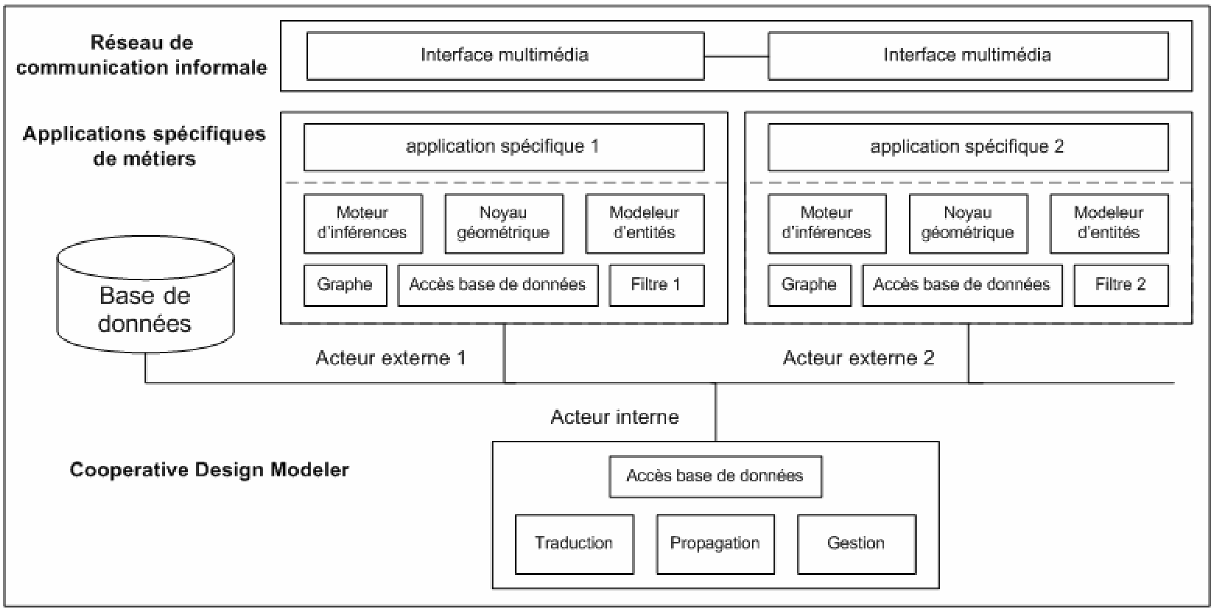


Figure 10 Architecture du système de CAID

Comme déjà présenté, le processus de conception intégrée est divisé en deux phases. Nous prenons en compte, dans notre application sur les meubles en bois, de trois domaines "métier" : assemblage, mécanique et usinage. Nous présentons ici l'interaction entre les acteurs pendant les deux phases:

4.1 Les phases de conception

Au début de la première phase, le designer, au sens français du terme, ou l'acteur concerné par la forme globale et l'esthétique du produit, telle la dimension, la texture, la couleur, etc., doit proposer un modèle conceptuel du produit. Le modèle conceptuel est normalement manipulé par un système de CAO. En conséquence, cet acteur peut le produire dans un format standard universel. Le résultat est donc la définition des surfaces fonctionnelles d'usage, ce qui clos la première phase.

6 Computer Aided Integrated Design = Conception Intégrée Assistée par Ordinateur

Dans notre étude, nous appliquons un format STEP[7]. Actuellement, nous demandons au technologue de prendre l'information initiale par ce fichier STEP pour le transformer dans l'environnement collaboratif, tel que en représenté dans la figure 11. Cette information initiale comporte la forme globale et les dimensions par défaut du produit. Celles-ci sont utilisées en tant que point de départ pour l'intégration de la connaissance dans la deuxième phase de conception.

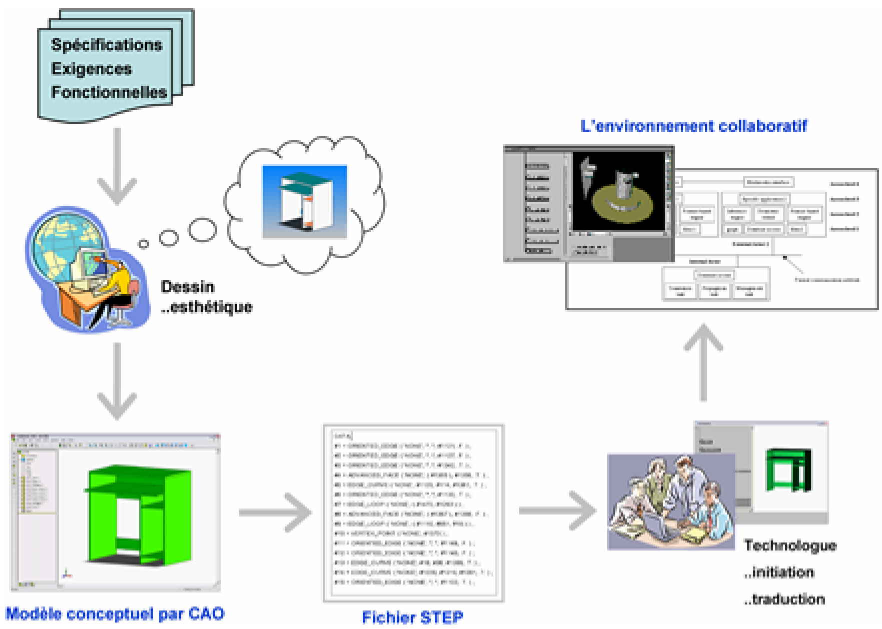


Figure 11 Le processus de passage de la 1ère à la 2ème phase de conception

A partir de l'initiation d'un projet de CoDeMo, le technologue demande à transformer le fichier STEP du modèle conceptuel. Nous prenons ici l'exemple d'un bureau d'ordinateur « DS-100 ». La figure 12 présente l'état initial dans la vue technologue et les vue communes : ossature et géométrie, alors que la figure 13 présente les vues après avoir transformé le fichier STEP par le processus de conception de la vue technologue. En fait, cette traduction est faite par l'acteur interne. Nous avons développé la méthode de traduction pour identifier les schémas du fichier STEP. L'acteur interne applique la méthode de traduction pour traduire le fichier STEP vers la base de données partagée et le représente sous la forme de modèle de produit (composants, liens, relations, et entités). Il utilise également la méthode de propagation pour transmettre une telle information aux vues communes et aux vues métiers correspondantes.

7 Standard for the Exchange of Product Model Data

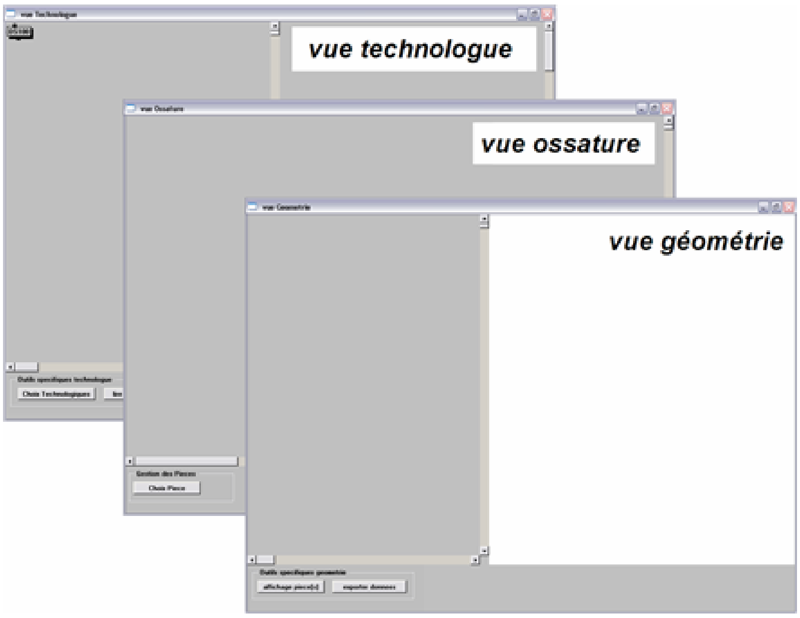


Figure 12 L'état initial de la vue technologue et des vue communes

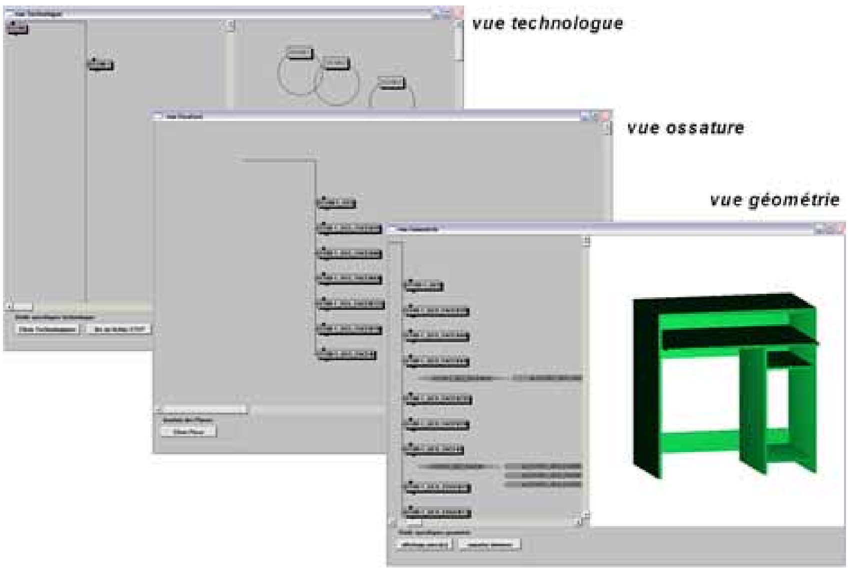


Figure 13 L'information initiale présentée après la transformation

Dans la vue géométrique, l'acteur interne peut établir automatiquement les entités préliminaires de contrainte géométrique qui affectent aux FRs du produit. Ces entités permettent aux acteurs d'identifier la structure de produit: quelle planche est en contact avec quelle autre, avec quelle surface, quelles planches sont parallèles ou perpendiculaires aux autres, ou quelles planches sont symétrique, etc.

La figure 14 montre des exemples de l'identification des entités des contraintes géométriques entre deux planches. Grâce aux données géométriques dans le fichier STEP, l'acteur interne vérifie la relation entre deux planches. Puis, il définit des entités entre des surfaces des planches suivant les directions des axis X, Y et Z. Avec ces définitions, l'acteur interne peut créer automatiquement des liens et la relation de contrainte géométrique entre les planches.

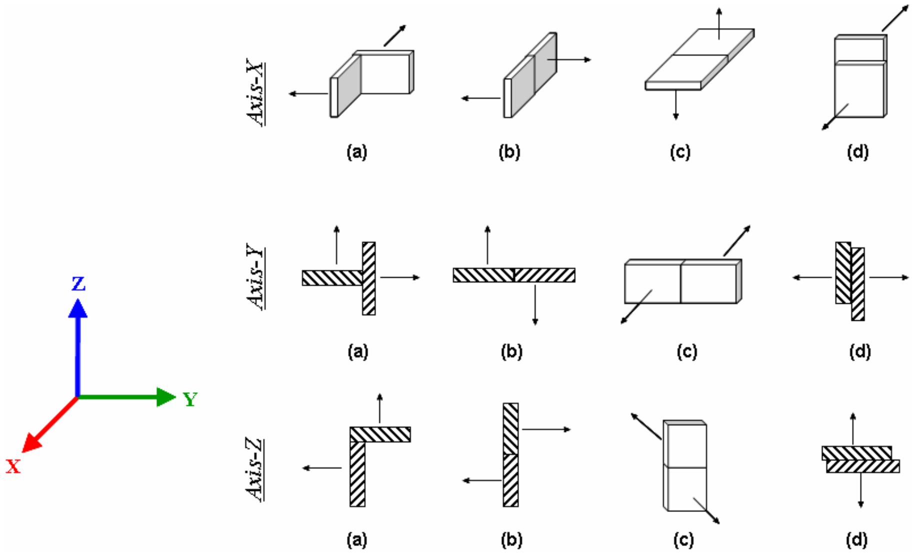


Figure 14 Exemples des entités de contrainte géométrique

Suite de la transformation par l'acteur interne, les vues métiers : assemblage, mécanique, et usinage, ont reçues l'information initiale du produit. Les figures suivantes présentent l'information initiale dans ces vues. Nous présenterons l'interaction entre les acteurs dans la session suivante.

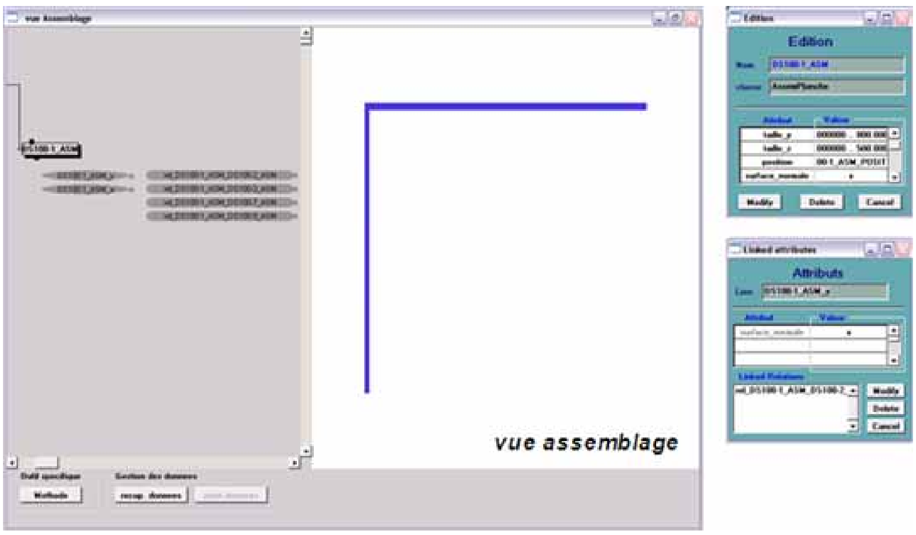


Figure 15 L'information initiale présentée dans la vue d'assemblage

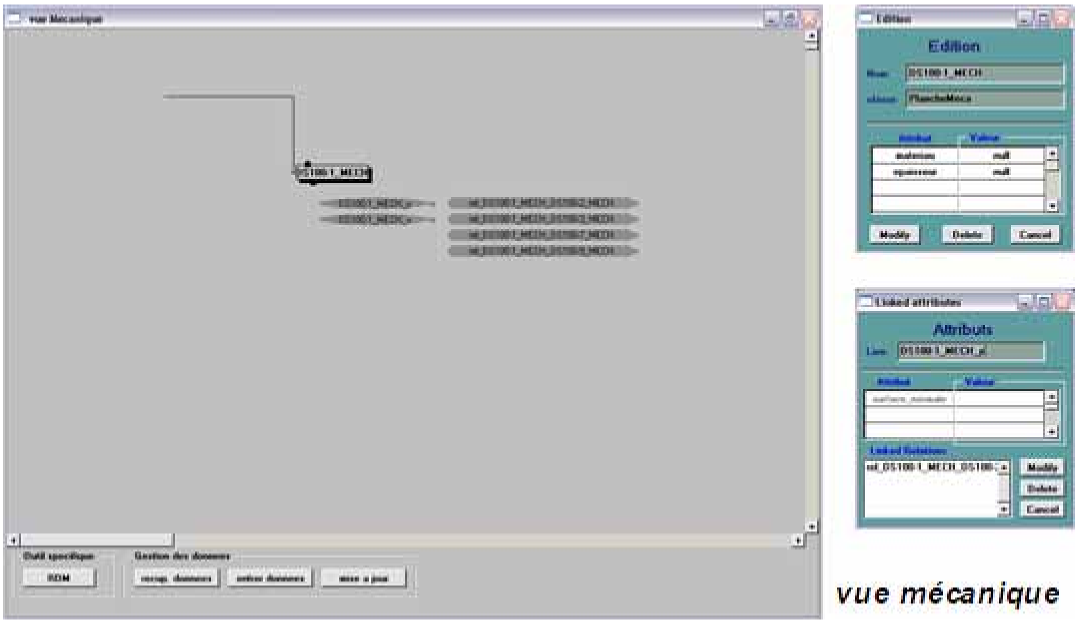


Figure 16 L'information initiale présentée dans la vue de mécanique

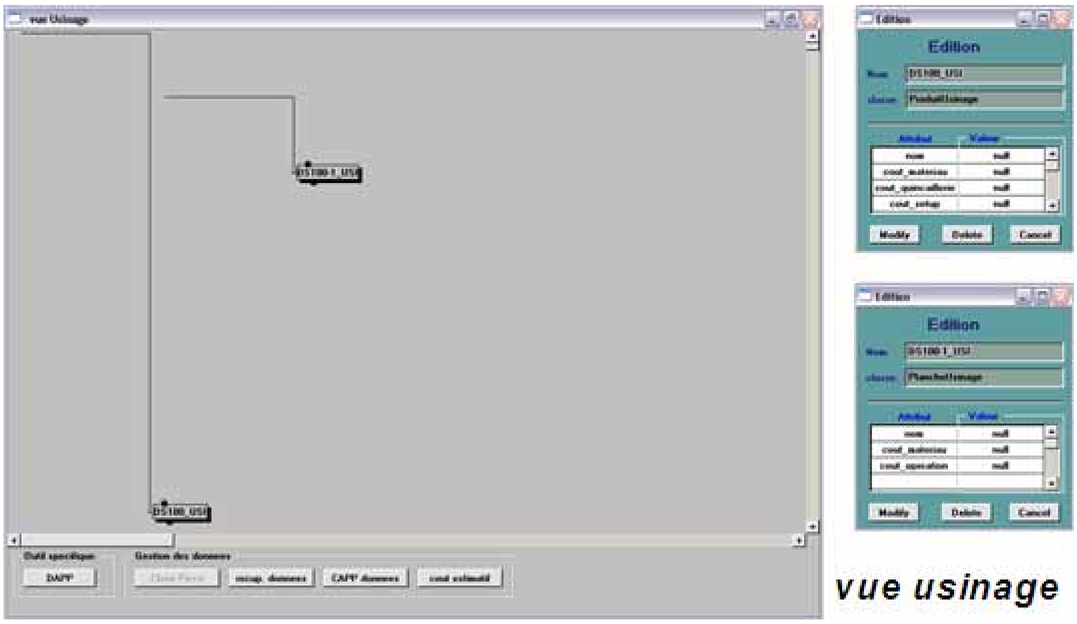


Figure 17 L'information initiale présentée dans la vue d'usinage

Chaque acteur s'occupe alors de tâches différentes mais le but commun est de contribuer pas à pas à établir de l'information, ajouter des contraintes au modèle produit. Nous postulons que les acteurs dans les vues métiers ont des expériences sur le problème et sont experts dans leur domaine de compétence.

Dans la vue d'assemblage, l'information initiale permet à l'assembleur de visualiser la vue d'ensemble du produit. Puisqu'il n'y a pas qu'une solution possible pour assembler entre deux pièces, CoDeMo crée un panneau pour afficher les solutions possibles. Ceci permet à l'assembleur de choisir une solution pour les planches comme représenté dans la figure 18.

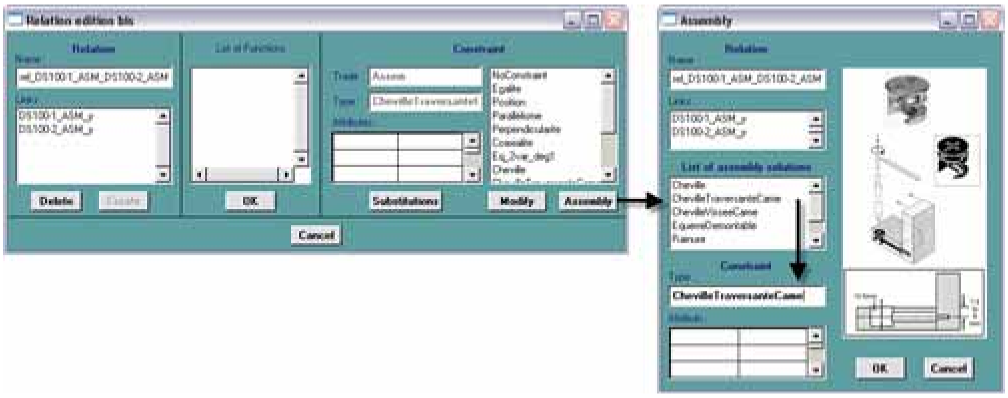


Figure 18 Choix d'une solution dans la vue d'assemblage

Dès que l'assembleur a choisi une solution, l'acteur interne va chercher une règle de production correspondante si elle existe et traduira cette décision vers les vues correspondantes, comme représenté figure 19. Cette traduction crée des instances d'entités ou des données pour les objets correspondants. Des caractéristiques d'une telle instance peuvent ne pas contenir initialement de valeur. Pourtant, celles-ci seront introduites dès que qu'un des acteurs a suffisamment d'information venant des autres acteurs ou par une évaluation.

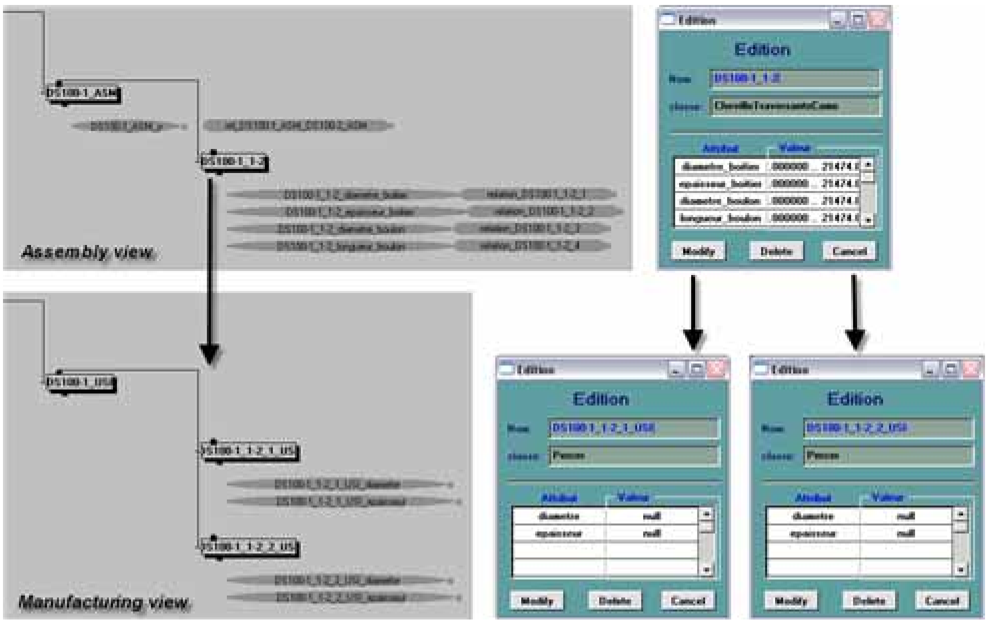
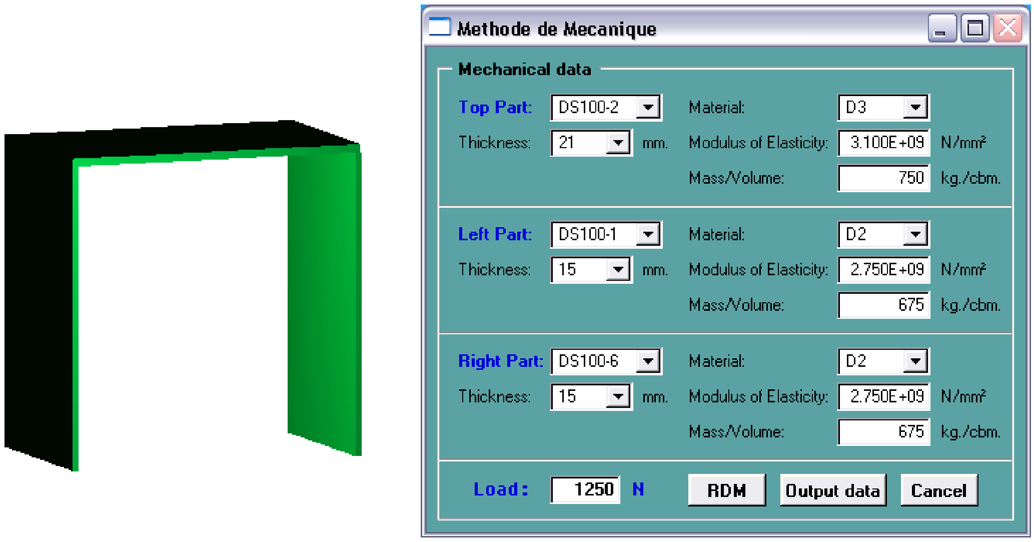


Figure 19 Exemple de la traduction et de la propagation d'une entité d'assemblage

Dans la vue de mécanique, CoDeMo crée une bibliothèque pour stocker l'information sur les matériaux disponibles, par exemple, types de matériaux, propriétés physiques et mécaniques de matériaux: densité, MOR[8], MOE[9], etc. L'objectif du mécanicien est de définir le type de matériau puis l'épaisseur appropriée pour des planches. CoDeMo permet au mécanicien d'ajouter des données sur le produit via un panneau de saisie de données comme présenté dans la figure 20.



(a) (b)

Figure 20 Panneau de saisir des données dans la vue de mécanique

Pour exécuter un test mécanique, CoDeMo applique l'outil qui s'appelle *RDM6* développé par Yves Debard [Debard 2000a, Debard 2000b], en tant qu'application spécifique du mécanicien. L'objectif de cet outil est de calculer les structures en appliquant la *RDM6* ou la méthode des éléments finis. Nous appliquons cet outil dans cette étude pour estimer la déflexion des planches.

Actuellement, le mécanicien demande à l'acteur interne de traduire les données en fichier neutre et en format compatible avec *RDM 6*. La figure 21 représente la simulation du test de déflexion, tandis que la figure 22 représente les résultats de la simulation pour cet exemple. Dès que le calcul est fait, le mécanicien doit transmettre la nouvelle information en fournissant les résultats au modèle produit. Il crée un fichier neutre qui permet à CoDeMo de comprendre et de traduire son information vers l'équipe de conception. La figure 23 présente la traduction et la propagation de la nouvelle information entre la vue mécanique et la vue assemblage.

[8] module de rupture

[9] module d'élasticité

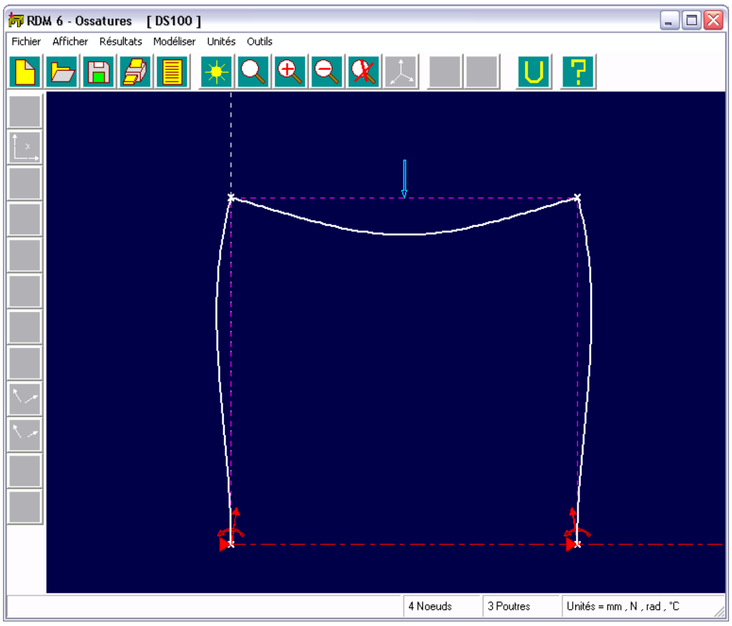


Figure 21 Simulation du test de déflexion

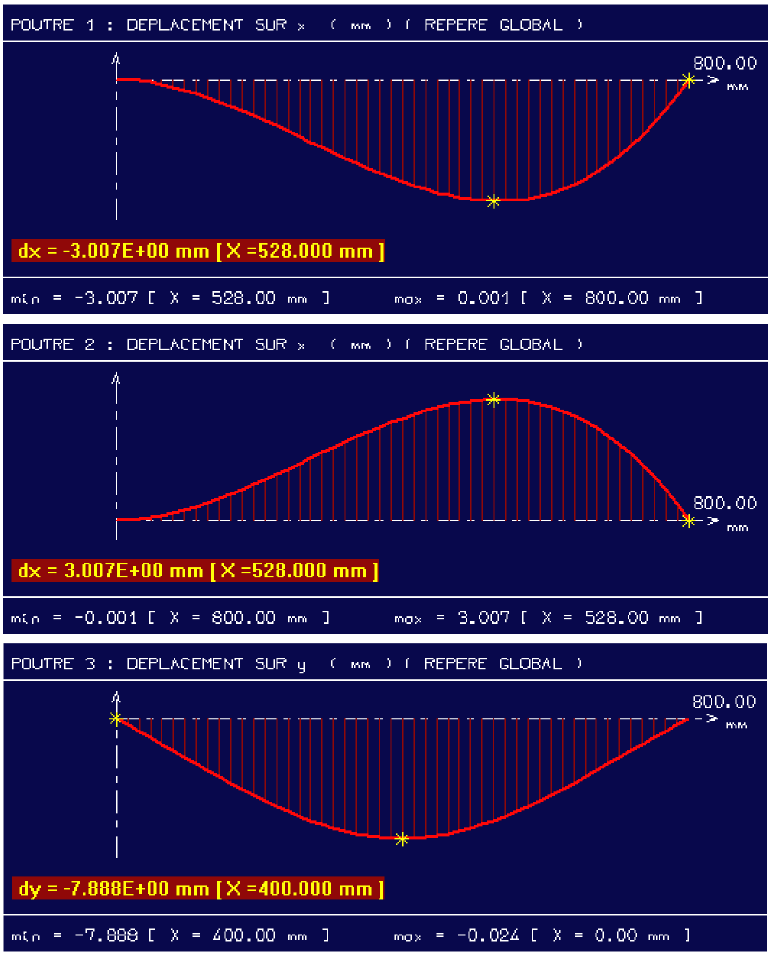


Figure 22 Résultats de la simulation

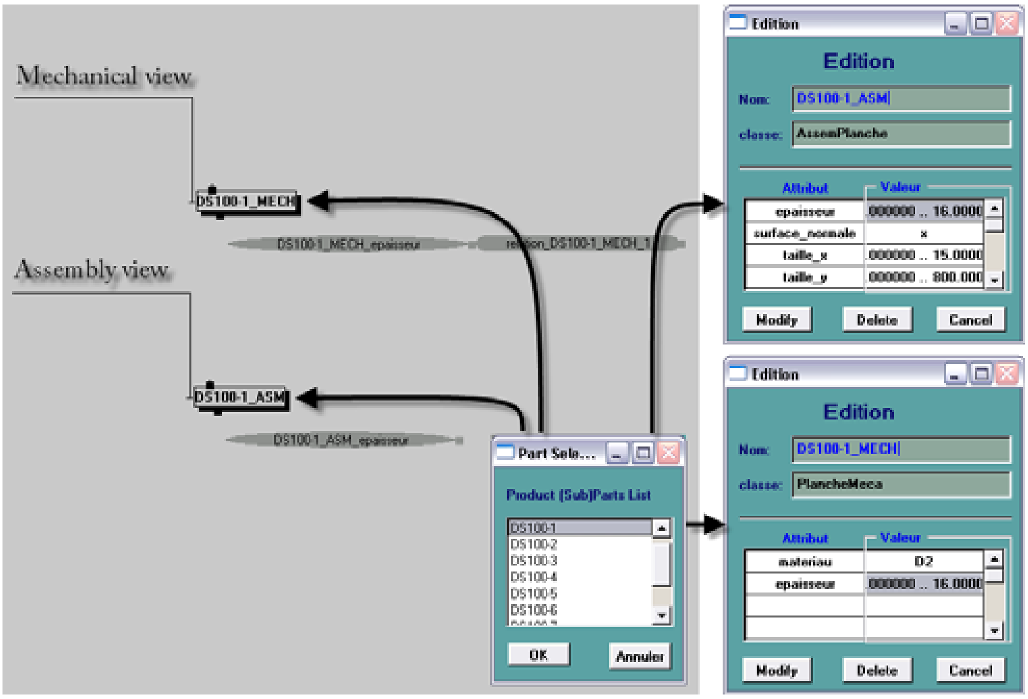


Figure 23 Exemple de la traduction et de la propagation dans la vue de mécanique

Dans la vue d'usinage, l'objectif est de planifier le processus de fabrication et d'estimer le coût. Le fabricant utilise une application spécifique « *DAPP* »[10] pour exécuter des tâches. CoDeMo recueilli des informations nécessaires et les transforme en fichier neutre pour être utilisable par *DAPP* comme représenté dans la figure 24.

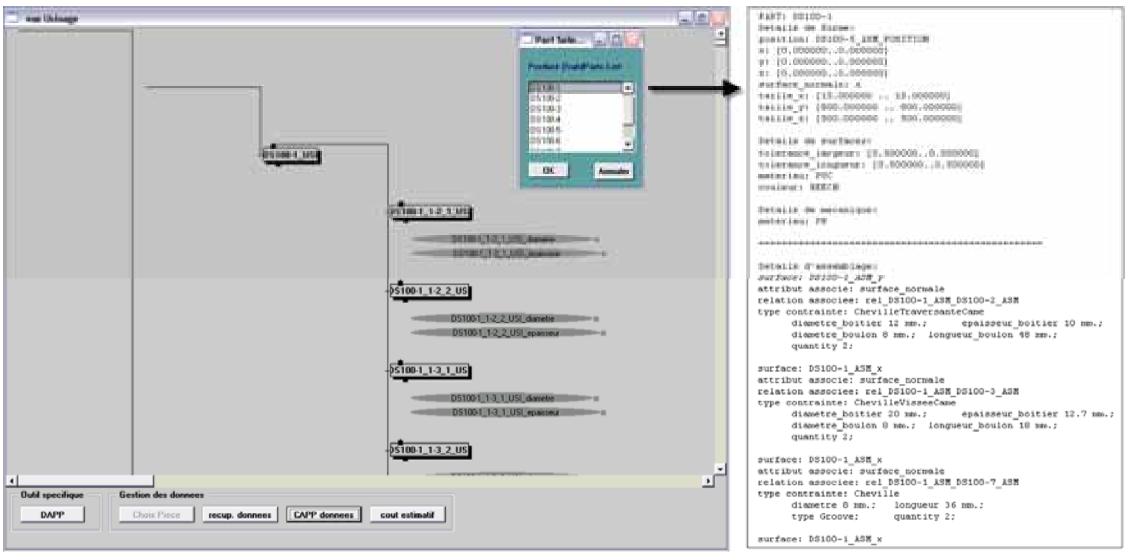


Figure 24 Exemple de la traduction et de la propagation dans la vue d'usinage

10 Database Application for Production Planning

Le fabricant rapporte une telle information à la base de données de DAPP via une interface des données de produit. Puis, il pourra évaluer la conception. Le lecteur peut se reporter pour plus information sur les fonctionnalités de DAPP dans notre thèse.

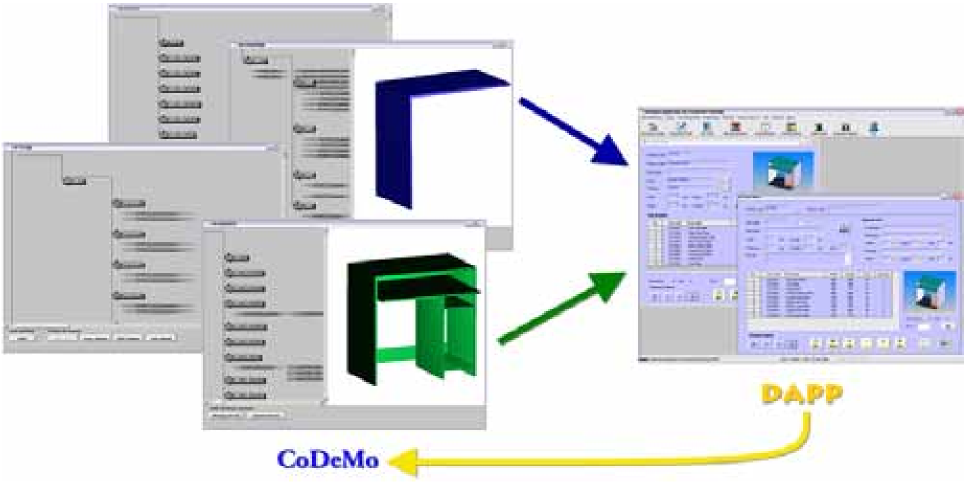


Figure 25 L'interaction entre CoDeMo et DAPP

Dès que l'évaluation de conception est réalisée, DAPP transfert les résultats, tels que: besoin de matériaux, coût de fabrication, coût des matériaux et des matières premières, temps de fabrication, etc. Pour réaliser cette tâche, DAPP crée un fichier textuel pour stocker des résultats sur l'évaluation dans un format que l'acteur interne peut comprendre. Ceci permet à CoDeMo de traduire les résultats dans le modèle produit. La figure 26 montre, par exemple, des résultats présentés dans la vue d'usinage.

Figure 26 Des valeurs de caractéristiques dans la vue d'usinage après la traduction

5. Conclusion

Les objectifs de cette étude étaient de développer un modeleur de conception pour un système de conception intégrée, de proposer une méthode pour réduire la complexité imaginaire dans le processus de conception, et de développer un processus de conception pour l'industrie de meubles réalisés en panneau de fibres ou de particules.

Un objectif principal de la conception intégrée est de réduire les itérations de conception en tenant compte des contraintes de différentes disciplines aussitôt que possible avant de prendre une décision. Pour réaliser cet objectif, les acteurs doivent transmettre leurs informations à l'équipe de conception dès que possible. Le concept de multi-représentation permet aux acteurs de présenter leurs informations dans l'environnement collaboratif tandis que le concept de multi-acteur et de multi-vue, permettent aux acteurs de dialoguer, de discuter, et de négocier sur le problème pendant le processus de conception. En outre, pour gérer la connaissance et l’information des acteurs différents, les méthodes de propagation et de traduction sont appliquées dans cette étude.

Afin de résoudre un problème de complexité, nous supposons dans cette étude que le problème de conception est une complexité imaginaire. En conséquence, nous proposons le système de conception intégrée, CoDeMo, pour introduire les acteurs autour d’une table dans une réunion virtuelle. Nous postulons que chaque acteur a des connaissances et des expériences sur des problèmes de conception et peut avoir un accès aux données existantes des problèmes de conception. Nous postulons également que les acteurs travaillent dans la notion de « juste besoin» pendant qu'ils résolvent les problèmes dans le processus de conception. Avec l'appui de la méthode et des modèles pour l'intégration, le système de conception intégrée permet à l'équipe de conception de réduire la complexité en résolvant les problèmes de la conception non-couplée, découplée, et faiblement couplée.

Dans cette étude, nous avons divisé le processus intégré de conception en deux phases. La première phase est principalement concernée par le designer. En deuxième phase, le technologue doit ensuite récupère l'information initiale en la transformant en modèle conceptuel du produit manipulé, à partir d'un système CAO. Les autres acteurs sont alors invités à participer au processus de conception pour transmettre leurs informations, contraintes, et points de vue à l'équipe de conception. Dans cette étude, nous avons pris en compte principalement trois domaines de compétence : assemblage, mécanique, et usinage. L’objectif de l'assembleur est de choisir les solutions d'assemblage appropriées; le mécanicien doit définir le matériau et l'épaisseur appropriés pour chaque planche, tandis que le fabricant doit évaluer le processus de fabrication et le coût. Chaque acteur de conception a différentes missions mais leurs informations sont toujours relatives.

Nous avons utilisé des entités et des règles de production. Une entité est utilisée pour décrire des éléments et des comportements du produit. Selon le concept du "mondes de conception", nous avons classifié les dispositifs dans trois sens : vernaculaire, véhiculaire, et universel. Le système permet aux acteurs de créer et d’utiliser de telles entités en donnant un accès à un moteur d’entités. Pour faire vivre et garder la cohérence de l'information, le

concept des règles de production est appliqué. Une règle de production est un élément de la connaissance qui est utilisée dans le processus de résolution des problèmes. Elle permet aux acteurs de partager et d'échanger leur information vers l'équipe. Des règles de production sont stockées dans le système sous une forme d'un fichier neutre. Pour permettre aux acteurs de conception de créer et d'employer une telle connaissance temporelle, le système doit leur donner un accès à un moteur d'inférence.

Nous avons appliqué le système de conception intégrée à l'industrie de meubles réalisés en panneau de fibres ou de particules. En raison du cycle de vie court et de rapide changement de ce type de meubles, il est difficile d'évaluer le processus de conception uniquement avec un modeleur de conception intégrée. En conséquent, le système de conception intégrée permet aux acteurs d'utiliser leurs applications spécifiques pour évaluer la conception. Nous avons présenté *RDM 6* pour évaluer la déflexion des planches dans la vue de mécanique, et également présenté *DAPP* dans la vue d'usinage pour évaluer principalement le coût de fabrication et planifier le processus de fabrication.

Bibliography

[Brissaud et al 1997] D. Brissaud, H. Paris, and S. Tichkiewitch, "Assisting designers in the forecasting of surfaces used for easier fixturing in the machining process," *Journal of Materials Processing technology*, vol. 65, pp. 26-33, 1997.

[Chapa Kasusky 1997] E. C. Chapa Kasusky, "Outils et structure pour la coopération formelle et informelle dans un contexte de conception holonique," in *Mécanique: Conception, géomécanique et matériaux*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 1997, pp. 132.

[Debard 2000a] Y. Debard, "RDM 6," 6.07 ed. Le Mans, 2000.

[Debard 2000b] Y. Debard, *Découvrir le calcul de structure avec RDM 5*. Le Mans, 2000.

[De Turck 2005] B. de Turck, "The competitiveness of the furniture industry," presented at The Future of the Furniture Industry in an Enlarged Europe, Brussels, Belgium, 2005.

[FFE 2006] FFE, "International: A Rosy Picture for 2006," in *Furniture & Furnishing Export International*, 2006.

[Gaucheron 2000] T. Gaucheron, "Intégration du recyclage en conception Le Modèle produit : Un outil descriptif et cognitif dans le processus de prise en compte du recyclage," in *Génie Industriel*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 2000, pp. 252.

[Heylighen 2002] F. Heylighen, "Complexity and Information Overload in Society: why increasing efficiency leads to decreasing control," 2002, pp. 44.

[Lu 2006] S. C.-Y. Lu, "Complexity in Collaborative Engineering," presented at International Consortium for Complexity Research (ICCR) Forum, Alberta, Canada, 2006.

[Mer et al 1995] S. Mer, A. Jeantet, and S. Tichkiewitch, *Les objets intermédiaires de la conception : modélisation et communication*. Paris, FRANCE: Europia, 1995.

[Mer 1998] S. Mer, "Les mondes et les outils de la conception. Pour une approche socio-technique de la conception de produit," in *Génie Industriel*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 1998, pp. 230.

[Radulescu 2005] B. Radulescu, "Développement d'un système de contrôle et de gestion des processus de conception intégrée : application à la traduction et la substitution au sein d'un modèle produit," in *Mécanique: Conception, géomécanique et matériaux*. Grenoble, FRANCE: Institut national polytechnique de Grenoble, 2005, pp. 234.

[Roucoules 1999] L. Roucoules, "Méthodes et connaissances : contribution au développement d'un environnement de conception intégrée," in *Mécanique*. Grenoble, FRANCE: INP de Grenoble, 1999, pp. 217.

[Roucoules and Tichkiewitch 2000] L. Roucoules and S. Tichkiewitch, "CoDE: A Cooperative Design Environment-- A New Generation of CAD Systems," *Concurrent Engineering*, vol. 8, pp. 263-280, 2000.

[Roucoules et al 2003] L. Roucoules, O. Salomons, and H. Paris, "Process planning as an integration of knowledge in the detailed design phase," *International Journal of Computer Integrated Manufacturing*, vol. 16, pp. 25-37, 2003.

[Salomone 1995] T. A. Salomone, *What every engineer should know about Concurrent Engineering*. New York: Marcel Dekker, 1995.

[Sky and Buchal 1999] R. W. E. Sky and R. O. Buchal, "Modeling and Implementing Concurrent Engineering in a Virtual Collaborative Environment," *Concurrent Engineering*, vol. 7, pp. 279-289, 1999.

[Sohlenius 1992] G. Sohlenius, "Concurrent Engineering," presented at 42nd General Assembly of CIRP, Aix-en-Provence, France, 1992.

[Suh 1990] N. P. Suh, *The principles of Design*. New York: Oxford University Press, 1990.

[Suh 2001] N. P. Suh, *Axiomatic Design Advances and Applications*. New York: Oxford University Press, 2001.

[Suh 2003] N. P. Suh, "A Theory of Complexity and Applications," 2003.

[Tichkiewitch 1990] S. Tichkiewitch, "Quelques problèms posés par la conception intégrée," in *Calcul Des Structures Et Intelligence Artificielle*, vol. 4, J. M. Fouet, L. P., and R. Ohayon, Eds. Paris: Pluralis, 1990, pp. 251-262.

[Tichkiewitch 1994] S. Tichkiewitch, "De la CFAO à la conception intégrée," *International Journal of CAD/CAM and Computer Graphics*, vol. 9, pp. 609-621, 1994.

[Tichkiewitch 1996] S. Tichkiewitch, "Specifications on integrated design methodology using a multi-view product model," presented at Engineering Systems Design and Analysis Conference, Montpellier, 1996.

[UEA 2005] UEA, "The World Furniture Industry: Production-Trade-Market," UEA: European Furniture Manufacturers Federation, 2005.

[UNECE 2005] UNECE, "European Forest Sector Outlook Study - Main Report," UNECE: United Nations Economic Commission for Europe, Geneva, Switzerland AE428/E, 2005.

Annex I

Acquisition of knowledge model in trade views

Examples of descriptive features for assembly solution

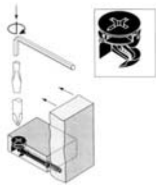
Features and characteristics

Dowel
- Type (Strand, Groove)
- Diameter
- Length
- Material (Wood, Metal)
- Maximum load


Screw
- Type (Tapping, Confirmat, Mounting, ...)
- Diameter
- Length
- Maximum load



Connector Joints
- Type (Minifix, Knock-down fitting, ...)
- Diameter of housing
- Length of housing
- Diameter of bolt
- Length of bolt
- Maximum load




Support
- Type (Plastic, Steel, ...)
- Diameter
- Length
- Material
- Maximum load


Grooving
- Type (Through, Distant)
- Width
- Depth
- Length

Examples of mechanical property for particleboard using in the mechanical view [23]

Name	Physical mechanical properties			Screw-holding	
	Modulus of Elasticity *(N/mm^2)*	***Modulus of Rupture*** *(N/mm^2)*	***Internal bond*** *(N/mm^2)*	***Face*** *(N)*	***Edge*** *(N)*
H-1	16.5	2400	0.90	1800	1325
H-2	20.5	2400	0.90	1900	1550
H-3	23.5	2750	1.00	2000	1550
M-1	11.0	1725	0.40	NS	NS
M-S	12.5	1900	0.40	900	800
M-2	14.5	2250	0.45	1000	900
M-3	16.5	2750	0.55	1100	1000
LD-1	30	550	0.10	400	NS
LD-2	5.0	1025	0.15	550	NS
PBU	11.0	1725	0.40	NS	NS
D-2	16.5	2750	0.55	NS	NS
D-3	19.5	3100	0.55	NS	NS

Examples of mechanical property for MDF using in the mechanical view [24]

Name	Physical mechanical properties			Screw-holding	
	Modulus of Elasticity *(N/mm^2)*	***Modulus of Rupture*** *(N/mm^2)*	***Internal bond*** *(N/mm^2)*	***Face*** *(N)*	***Edge*** *(N)*
110	14.0	1400	0.30	780	670
120	14.0	1400	0.50	875	775
130	24.0	2400	0.60	1100	875
140	24.0	2400	0.75	1325	1000
150	31.0	3100	0.90	1400	1200
160	31.0	3100	1.05	1555	1335

[23] From the standard requirements of ANSI A208.1-1999

[24] From the standard requirements of ANSI A208.2-2002

Production rules for feature Tourillon in QTrans file

```
Component_Name
        Tourillon Assem name

Traduction
        Component Percer Usinage name_1_USI
        Component Percer Usinage name_2_USI
        Link name diametre name_diametre
        Link name longueur name_longueur
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Link name_2_USI diametre name_2_USI_diametre
        Link name_2_USI epaisseur name_2_USI_epaisseur
        Relation name_diametre name name_1_USI_diametre name_1_USI relation_name_1
        Relation name_longueur name name_1_USI_epaisseur name_1_USI relation_name_2
        Relation name_diametre name name_2_USI_diametre name_2_USI relation_name_3
        Relation name_longueur name name_2_USI_epaisseur name_2_USI relation_name_4
@
```

Production rules for feature TourillonTraversanteCame in QTrans file

```
Component_Name
        TourillonTraversanteCame Assem name

Traduction
        Component Percer Usinage name_1_USI
        Component Percer Usinage name_2_USI
        Link name diametre_boitier name_diametre_boitier
        Link name epaisseur_boitier name_epaisseur_boitier
        Link name diametre_boulon name_diametre_boulon
        Link name longueur_boulon name_longueur_boulon
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Link name_2_USI diametre name_2_USI_diametre
        Link name_2_USI epaisseur name_2_USI_epaisseur
        Relation name_diametre_boitier  name name_1_USI_diametre name_1_USI
        relation_name_1
        Relation name_epaisseur_boitier name name_1_USI_epaisseur name_1_USI
        relation_name_2
        Relation name_diametre_boulon name name_2_USI_diametre name_2_USI
        relation_name_3
        Relation name_longueur_boulon name name_2_USI_epaisseur name_2_USI
        relation_name_4
@
```

Production rules for feature TourillonVisseeCame in QTrans file

Component_Name
    TourillonVisseeCame Assem name

Traduction
    Component Percer Usinage name_1_USI
    Component Percer Usinage name_2_USI
    Link name diametre_boitier name_diametre_boitier
    Link name epaisseur_boitier name_epaisseur_boitier
    Link name diametre_boulon name_diametre_boulon
    Link name longueur_boulon name_longueur_boulon
    Link name_1_USI diametre name_1_USI_diametre
    Link name_1_USI epaisseur name_1_USI_epaisseur
    Link name_2_USI diametre name_2_USI_diametre
    Link name_2_USI epaisseur name_2_USI_epaisseur
    Relation name_diametre_boitier name name_1_USI_diametre name_1_USI relation_name_1
    Relation name_epaisseur_boitier name name_1_USI_epaisseur name_1_USI relation_name_2
    Relation name_diametre_boulon name name_2_USI_diametre name_2_USI relation_name_3
    Relation name_longueur_boulon name name_2_USI_epaisseur name_2_USI relation_name_4
@

Production rules for feature Rainure in QTrans file

Component_Name
    Rainure Assem name

Traduction
    Component Rainurer Usinage name_1_USI
    Link name largeur name_largeur
    Link name epaisseur name_epaisseur
    Link name distance name_distance
    Link name_1_USI largeur name_1_USI_largeur
    Link name_1_USI epaisseur name_1_USI_epaisseur
    Link name_1_USI distance name_1_USI_distance
    Relation name_largeur name name_1_USI_largeur name_1_USI relation_name_1
    Relation name_epaisseur name name_1_USI_epaisseur name_1_USI relation_name_2
    Relation name_distance name name_1_USI_distance name_1_USI relation_name_3
@

Production rules for feature Support in QTrans file

```
Component_Name
        Support Assem name

Traduction
        Component Percer Usinage name_1_USI
        Link name diametre name_diametre
        Link name longueur name_longueur
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Relation name_diametre name name_1_USI_diametre name_1_USI relation_name_1
        Relation name_longueur name name_1_USI_epaisseur name_1_USI relation_name_2
@
```

Production rules for feature Vis in QTrans file

```
Component_Name
        Vis Assem name

Traduction
        Component Percer Usinage name_1_USI
        Component Percer Usinage name_2_USI
        Link name diametre name_diametre
        Link name longueur name_longueur
        Link name_1_USI diametre name_1_USI_diametre
        Link name_1_USI epaisseur name_1_USI_epaisseur
        Link name_2_USI diametre name_2_USI_diametre
        Link name_2_USI epaisseur name_2_USI_epaisseur
        Relation name_diametre name name_1_USI_diametre name_1_USI relation_name_1
        Relation name_longueur name name_1_USI_epaisseur name_1_USI relation_name_2
        Relation name_diametre name name_2_USI_diametre name_2_USI relation_name_3
        Relation name_longueur name name_2_USI_epaisseur name_2_USI relation_name_4
@
```

Production rules for feature Tourillon between mechanical and assembly view

```
Component_Attribute
    PlancheMeca Meca name_MECH
    Tourillon Assem name_ASM

Traduction
    Attribute name_MECH materiau Char materiau_planche
    Attribute name_MECH epaisseur Float epaisseur_planche
    Attribute name_ASM epaisseur Float epaisseur_planche
    Link name_MECH epaisseur name_MECH_epaisseur
    Link name_ASM epaisseur name_ASM_epaisseur
    Relation name_MECH_epaisseur name_MECH name_ASM_epaisseur name_ASM
    relation_MECH_1
@
```

Production rules for characteristics of feature Tourillon in QTrans file

```
Component_Name
    Tourillon Assem name

SubComponent_Name
    SubComponent Percer Usinage name_1_USI
    SubComponent Percer Usinage name_2_USI

Traduction
    Attribute name type Char type_tourillon
    Attribute name diametre Float diametre_tourillon
    Attribute name longueur Float longueur_tourillon
    Attribute name quantity Int qty_tourillon
    Attribute name_1_USI diametre Float diametre1_tourillon
    Attribute name_1_USI epaisseur Float epaisseur1_tourillon
    Attribute name_2_USI diametre Float diametre2_tourillon
    Attribute name_2_USI epaisseur Float epaisseur2_tourillon
@
```

Production rules for characteristics of feature TourillonTraversanteCame in QTrans file

Component_Name
TourillonTraversanteCame Assem name

SubComponent_Name
SubComponent Percer Usinage name_1_USI
SubComponent Percer Usinage name_2_USI

Traduction
Attribute name diametre_boitier Float dia_boit_tourillonTC
Attribute name epaisseur_boitier Float epais_boit_tourillonTC
Attribute name diametre_boulon Float dia_boul_tourillonTC
Attribute name longueur_boulon Float epais_boul_tourillonTC
Attribute name quantity Int qty_tourillonTC
Attribute name_1_USI diametre Float dia_boit_tourillonTC
Attribute name_1_USI epaisseur Float epaisseur1_tourillonTC
Attribute name_2_USI diametre Float dia_boul_tourillonTC
Attribute name_2_USI epaisseur Float epaisseur2_tourillonTC
@

Production rules for characteristics of feature TourillonVisseeCame in QTrans file

Component_Name
TourillonVisseeCame Assem name

SubComponent_Name
SubComponent Percer Usinage name_1_USI
SubComponent Percer Usinage name_2_USI

Traduction
Attribute name diametre_boitier Float dia_boit_tourillonVC
Attribute name epaisseur_boitier Float epais_boit_tourillonVC
Attribute name diametre_boulon Float dia_boul_tourillonVC
Attribute name longueur_boulon Float epais_boul_tourillonVC
Attribute name quantity Int qty_tourillonVC
Attribute name_1_USI diametre Float dia_boit_tourillonVC
Attribute name_1_USI epaisseur Float epaisseur1_tourillonVC
Attribute name_2_USI diametre Float dia_boul_tourillonVC
Attribute name_2_USI epaisseur Float epaisseur2_tourillonVC
@

Production rules for characteristics of feature Rainure in QTrans file

Component_Name
    Rainure Assem name

SubComponent_Name
    SubComponent Rainurer Usinage name_2_USI

Traduction
    Attribute name largeur Float largeur_rainure
    Attribute name epaisseur Float epaisseur_rainure
    Attribute name distance Float distance_rainure
    Attribute name_1_USI largeur Float largeur_rainure
    Attribute name_1_USI epaisseur Float epaisseur_rainure
    Attribute name_1_USI distance Float distance_rainure
@

Production rules for characteristics of feature Support in QTrans file

Component_Name
    Support Assem name

SubComponent_Name
    SubComponent Percer Usinage name_1_USI

Traduction
    Attribute name type Char type_support
    Attribute name diametre Float diametre_support
    Attribute name longueur Float longueur_support
    Attribute name quantity Int qty_support
    Attribute name_1_USI diametre Float diametre_support
    Attribute name_1_USI epaisseur Float epaisseur_support
@

Production rules for characteristics of feature Vis in QTrans file

```
Component_Name
        Vis Assem name

SubComponent_Name
          SubComponent Percer Usinage name_1_USI
          SubComponent Percer Usinage name_2_USI

Traduction
        Attribute name type Char type_vis
        Attribute name diametre Float diametre_vis
        Attribute name longueur Float longueur_vis
        Attribute name quantity Int qty_vis
        Attribute name_1_USI diametre Float diametre1_vis
        Attribute name_1_USI epaisseur Float epaisseur1_vis
        Attribute name_2_USI diametre Float diametre2_vis
        Attribute name_2_USI epaisseur Float epaisseur2_vis
@
```

Annex II

Integrated design system

Example of an output file from CoDeMo

PART: DS100-1

Details de forme:
position: DS100-5_ASM_POSITION
x: [0.000000..0.000000]
y: [0.000000..0.000000]
z: [0.000000..0.000000]
surface_normale: x
taille_x: [15.000000 .. 15.000000]
taille_y: [800.000000 .. 800.000000]
taille_z: [500.000000 .. 500.000000]

Details de surfaces:
tolerance_largeur: [0.500000..0.500000]
tolerance_longueur: [0.500000..0.500000]
materiau: PVC
couleur: BEECH

Details de mecanique:
materiau: PB

Details d'assemblage:
surface: DS100-1_ASM_y
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-2_ASM
type contrainte: TourillonTraversanteCame
    diametre_boitier 12 mm.;   epaisseur_boitier 10 mm.;
    diametre_boulon 8 mm.;   longueur_boulon 48 mm.;
    quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-3_ASM
type contrainte: TourillonVisseeCame
    diametre_boitier 20 mm.;   epaisseur_boitier 12.7 mm.;
    diametre_boulon 8 mm.;   longueur_boulon 18 mm.;
    quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-7_ASM
type contrainte: Tourillon
    diametre 8 mm.;   longueur 36 mm.;
    type Groove;   quantity 2;

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-9_ASM
type contrainte: Rainure
largeur 4 mm.; profondeur 5 mm.; distance 280 mm.;

Details d'usinage:
surface: DS100-1_ASM_y
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-2_ASM
type contrainte: TourillonTraversanteCame

percage1:	diameter 12 mm.;	profondeur 10.25 mm.;
type:	non-debouchant;	
percage2:	diametre 8 mm.;	profondeur 34 mm.;
type:	debouchant;	

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-3_ASM
type contrainte: TourillonVisseeCame

percage1:	diametre 20 mm.;	profondeur 13 mm.;
type:	non-debouchant;	
percage2:	diametre 8 mm.;	profondeur 10 mm.;
type:	non-debouchant;	

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-7_ASM
type contrainte: Tourillon

percage1:	diametre 8 mm.;	profondeur 24 mm.;
type:	non-debouchant;	
percage2:	diametre 8 mm.;	profondeur 12 mm.;
type:	non-debouchant;	

surface: DS100-1_ASM_x
attribut associe: surface_normale
relation associee: rel_DS100-1_ASM_DS100-9_ASM
type contrainte: Rainure

rainurage:	largeur 4 mm.;	profondeur 5 mm.;	distance 280 mm.;
type:	non-traversant;		

Controlled document – Production order

Production Order

Purchasing Order No. CN-001	Production Order No. PN-002
Product Code: DC100	Ordered Date: 17/07/2007
Product Name: COMPUTER DESK	Starting Date: 26/07/2007
Product Color: Beech Quantity: 200 Unit(s)	Finished Date: 30/07/2007

No.	Part Code	Part Name	Width	Length	Thickness	Quantity	Remark
1	DC100-01	Left-Side Plate	500	800	15	206	
2	DC100-02	Right-Side Plate	500	800	15	206	
3	DC100-03	Vertical-Center Plate	485	650	15	206	
4	DC100-04	Back-Upper Plate	280	770	3	206	
5	DC100-05	Back-Lower Plate	100	770	15	206	
6	DC100-06	Upper-Fixed Plate	235	485	15	206	
7	DC100-07	Lower-Fixed Plate	235	485	15	206	
8	DC100-08	Horizontal-Center Plate	600	770	15	206	
9	DC100-09	Top Plate	500	800	21	206	

Note: ______________________________

Printing Date: 26 juillet 2007 Page: 1

Controlled document – Job order

G-SCOP

Job Order

Production Order No.	PN-002	Part Code:	DC100-01
Product Code:	DC100	Part Name:	Left-Side Plate
Product Name:	COMPUTER DESK	Part Size:	500 x 800 x 15 mm.
Color:	Beech	Quantity:	206 Piece(s)
Quantity:	200 Unit(s)	Remark:	

Job Order No.	JN-011
Date:	26/07/2007
Starting Date:	26/07/2007
Finished Date:	30/07/2007

No.	Work Center	Process Name	Remark
1	Cutting	Cutting	Tolerance +/- 0.5 mm.
2	Grooving	Distant Grooving	Groove 5 x 5 mm., 12 mm. from the edge for the Back-Upper Plate
3	Banding	Straight-Banding	Banding on the edges of length
4	Drilling	Drilling	2 x (6 mm.) for the Back-Lower Plate, 2 x (15 mm.) for the Wide Plate, 2 x (15 mm.) for the Top Plate
5	Support	Finishing	Label part number (1)

Note:

Printed Date: 26 juillet 2007

Page: 1

Materials requirement

Controlled document – Materials requirement by Product

Materials Requirement by Product

Printed Date: 26 juillet 2007

Materials requirement by product: DC100 Quantity: 100.00 Unit(s)

Part ID	Part name	Size	Quantity	Remark
401001	Bugle Head Screw 6*1	-	400	-
401005	Bugle Head Screw 8*1	-	200	-
401012	Taper Head Screw 4*20	-	800	-
402013	Minifix KD Fitting1/2"	-	400	-
403001	Compressed Dowel Pins 3/8" x 2"	-	400	-
405005	Desk pack	-	100	-

Note: ______________________________

Database Application for Production Planning Page: 1

Controlled document – Materials requirement by Purchasing Order

G-SCOP SCIENCES POUR LA CONCEPTION, L'OPTIMISATION ET LA PRODUCTION

Materials Requirement by Purchasing Order

Printed Date: 26 juillet 2007

Materials requirement by purchasing order no. CN-001 Ordered Date: 10 juillet 2006

Customer: Domaine Universitaire

Part ID	Part name	Size	Quantity	Remark
401008	Bugle Head Screw 8*1-1/2	-	400	-
401010	Bugle Head Screw 8*2	-	500	-
402013	Minifix KD Fitting1/2"	-	400	-
403004	Compressed Dowel Pins 1/4" x 1-1/2"	-	400	-
403005	Fluted Dowel Pins - 1/4" x 1-1/2"	-	1,200	-
405002	CD pack	-	100	-
405009	Foam 1"	-	200	-
401001	Bugle Head Screw 6*1	-	800	-
401005	Bugle Head Screw 8*1	-	400	-
401012	Taper Head Screw 4*20	-	1,600	-
402013	Minifix KD Fitting1/2"	-	800	-
403001	Compressed Dowel Pins 3/8" x 2"	-	800	-
405005	Desk pack	-	200	-

Note: ______________________________

Database Application for Production Planning Page: 1

Controlled document – Materials requirement by Purchasing Order, separated by item

G-SCOP

Materials Requirement by Purchasing Order

Printed Date: 26 juillet 2007

Materials requirement by purchasing order no. CN-001 Ordered Date: 10 juillet 2006

Customer: Domaine Universitaire Delivery Date: 31 juillet 2007

Product Code: CD-003 Product Name: CD Cabinet Color: Cherry Quantity: 100 Unit(s)

Part Code	Part Name	Size	Quantity	Remark
405009	Foam 1"	–	200	-
405002	CD pack	–	100	-
403005	Fluted Dowel Pins - 1/4" x	–	1,200	-
403004	Compressed Dowel Pins 1	–	400	-
402013	Minifix KD Fitting1/2"	–	400	-
401010	Bugle Head Screw 8*2	–	500	-
401008	Bugle Head Screw 8*1-1/2	–	400	-

Product Code: DC100 Product Name: COMPUTER DESK Color: Beech Quantity: 200 Unit(s)

Part Code	Part Name	Size	Quantity	Remark
405005	Desk pack	–	200	-
403001	Compressed Dowel Pins 3	–	800	-
402013	Minifix KD Fitting1/2"	–	800	-
401012	Taper Head Screw 4*20	–	1,600	-
401005	Bugle Head Screw 8*1	–	400	-
401001	Bugle Head Screw 6*1	–	800	-

Note:

Database Application for Production Planning Page: 1

Time estimation

Report of time estimation – Setup time

Time Estimation: Summary of Setup Time **Product: DC100**

Part Code	Part Name	Cutting	Distant Grooving	Drilling	Finishing	Straight-Banding	Through Grooving	Total
DC100-01	Left-Side Plate	10	10	15		10		45
DC100-02	Right-Side Plate	10	10	15		10		45
DC100-03	Vertical-Center Plate	10		15		10		35
DC100-04	Back-Upper Plate	10						10
DC100-05	Back-Lower Plate	10		15		10		35
DC100-06	Upper-Fixed Plate	10		15		10		35
DC100-07	Lower-Fixed Plate	10		15		10		35
DC100-08	Horizontal-Center Plate	10		15		10		35
DC100-09	Top Plate	10		15		10	10	45
Total		90	20	120		80	10	320

Database Application for Production Planning

Report of time estimation – Operation time

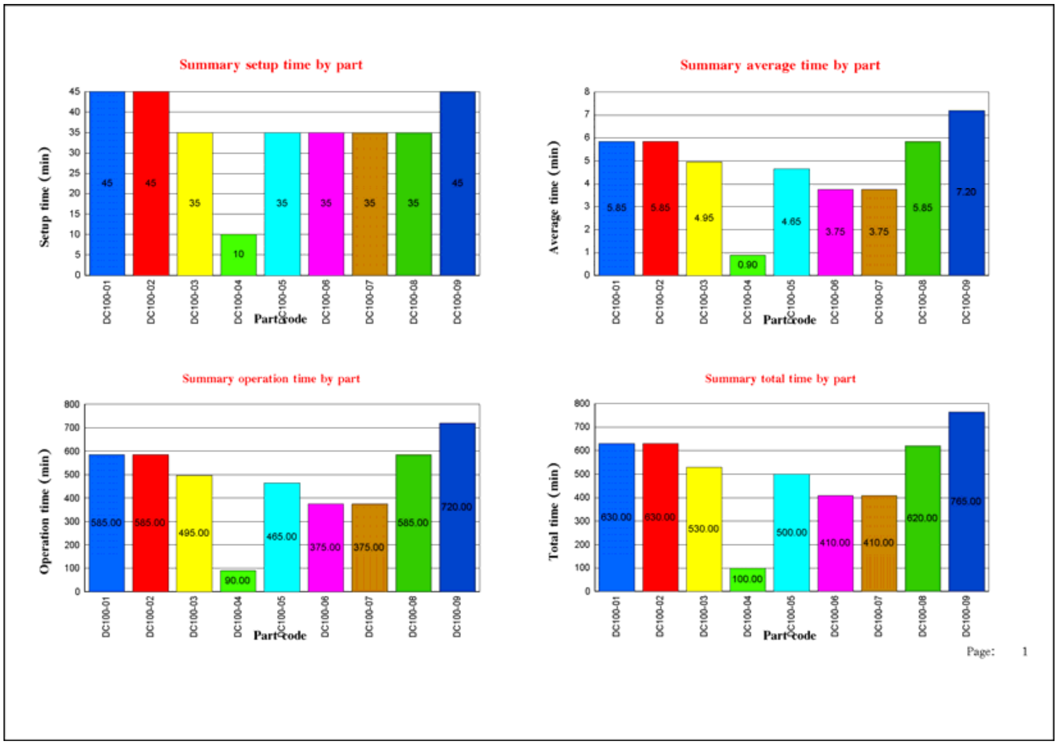
*** Time Estimation: Summary of OperationTime** **Product: DC100**

Part Code	Part Name	Cutting	Distant Grooving	Drilling	Finishing	Straight-Banding	Through Grooving	Total
DC100-01	Left-Side Plate	0.90	0.68	1.20	1.88	1.20		5.85
DC100-02	Right-Side Plate	0.90	0.68	1.20	1.88	1.20		5.85
DC100-03	Vertical-Center Plate	0.90		1.20	1.65	1.20		4.95
DC100-04	Back-Upper Plate	0.90						0.90
DC100-05	Back-Lower Plate	0.75		1.20	1.50	1.20		4.65
DC100-06	Upper-Fixed Plate	0.68		0.90	1.13	1.05		3.75
DC100-07	Lower-Fixed Plate	0.68		0.90	1.13	1.05		3.75
DC100-08	Horizontal-Center Plate	0.90		1.35	2.10	1.50		5.85
DC100-09	Top Plate	1.20		1.50	2.25	1.50	0.75	7.20
Total		7.80	1.35	9.45	13.50	9.90	0.75	42.75

* This summary operation time is calculate per 1 unit.

Database Application for Production Planning

Summary chart of time estimation represented by part – page 1



Summary chart of time estimation represented by part – page 2

Time Estimation: Represented by Part

Printed Date: 27 juillet 2007

Product Code: DC100 Product Name: COMPUTER DESK Quantity: 100 Unit(s)

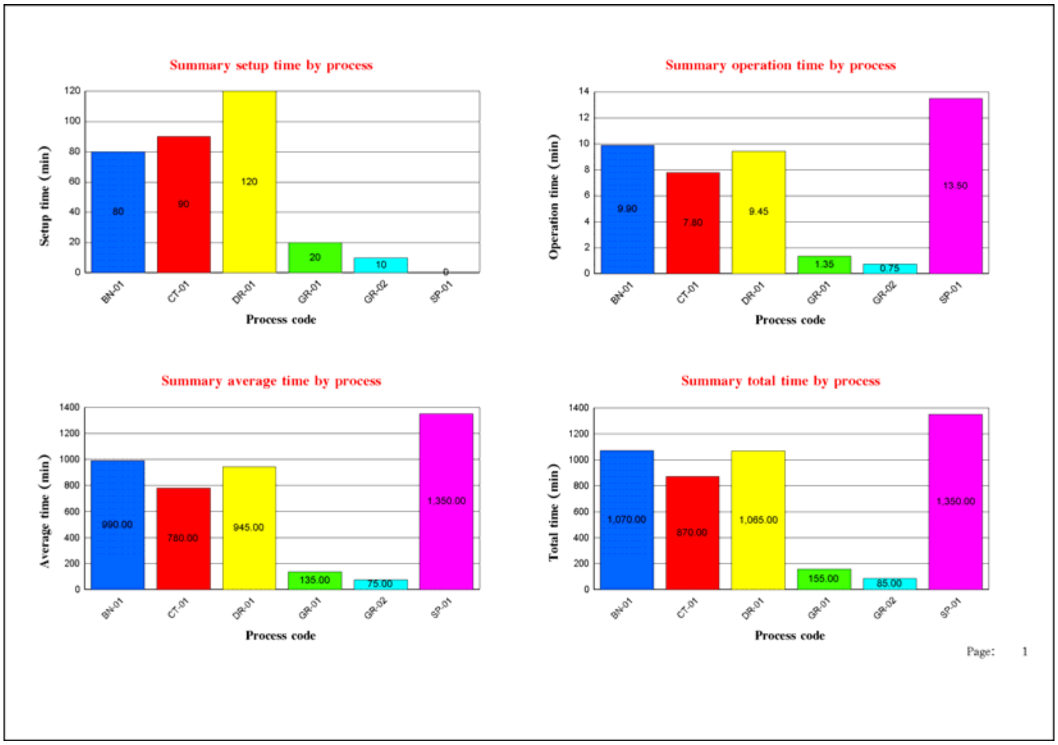
Part Code	Part Name	Setup time	Average time	Quantity	Operation time	Total time
DC100-01	Left-Side Plate	45	5.85	100	585.00	630.00
DC100-02	Right-Side Plate	45	5.85	100	585.00	630.00
DC100-03	Vertical-Center Plate	35	4.95	100	495.00	530.00
DC100-04	Back-Upper Plate	10	.90	100	90.00	100.00
DC100-05	Back-Lower Plate	35	4.65	100	465.00	500.00
DC100-06	Upper-Fixed Plate	35	3.75	100	375.00	410.00
DC100-07	Lower-Fixed Plate	35	3.75	100	375.00	410.00
DC100-08	Horizontal-Center Plate	35	5.85	100	585.00	620.00
DC100-09	Top Plate	45	7.20	100	720.00	765.00

Summary: Setup time: 320 min(s) Average time: 43 min(s)/unit

Operation time: 4,275 min(s) Total time: 4,595 min(s)

Database Application for Production Planning Page: 2

Summary chart of time estimation represented by process – page 1



Summary chart of time estimation represented by process – page 2

Time Estimation: Represented by Process

Printed Date: 27 juillet 2007

Product Code: DC100 Product Name: COMPUTER DESK Quantity: 100 Unit(s)

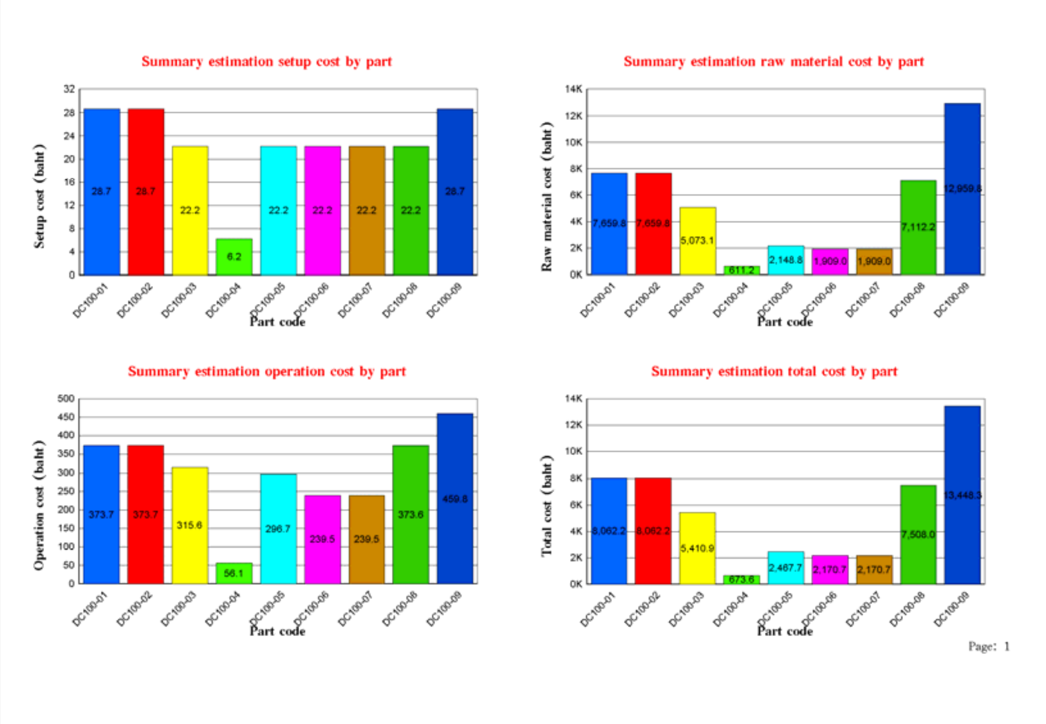
Process Code	Process Name	Setup time	Average time	Quantity	Operation time	Total time
BN-01	Straight-Banding	80	9.90	100	990.00	1,070.00
CT-01	Cutting	90	7.80	100	780.00	870.00
DR-01	Drilling	120	9.45	100	945.00	1,065.00
GR-01	Distant Grooving	20	1.35	100	135.00	155.00
GR-02	Through Grooving	10	0.75	100	75.00	85.00
SP-01	Finishing	0	13.50	100	1,350.00	1,350.00

Summary:	Setup time:	320.00	min(s)	Average time:	42.75	min(s)/unit
	Operation time:	4,275.00	min(s)	Total time:	4,595.00	min(s)

Database Application for Production Planning Page: 2

Cost estimation

Summary chart of cost estimation represented by part – page 1



Summary of cost estimation represented by part – page 2

Summary report of estimated production cost Print Date: 29 juillet 2007

Product: DC100 Product Name: COMPUTER DESK Quantity: 100 Units

Part Code	Part Name	Raw Meterial Cost	Setup cost	Operation cost	Total cost
DC100-01	Left-Side Plate	7,659.76	28.69	373.75	8,062.20
DC100-02	Right-Side Plate	7,659.76	28.69	373.75	8,062.20
DC100-03	Vertical-Center Plate	5,073.08	22.21	315.64	5,410.93
DC100-04	Back-Upper Plate	611.20	6.24	56.12	673.55
DC100-05	Back-Lower Plate	2,148.82	22.21	296.72	2,467.75
DC100-06	Upper-Fixed Plate	1,909.03	22.21	239.50	2,170.74
DC100-07	Lower-Fixed Plate	1,909.03	22.21	239.50	2,170.74
DC100-08	Horizontal-Center Plate	7,112.22	22.21	373.57	7,507.99
DC100-09	Top Plate	12,959.77	28.69	459.80	13,448.26

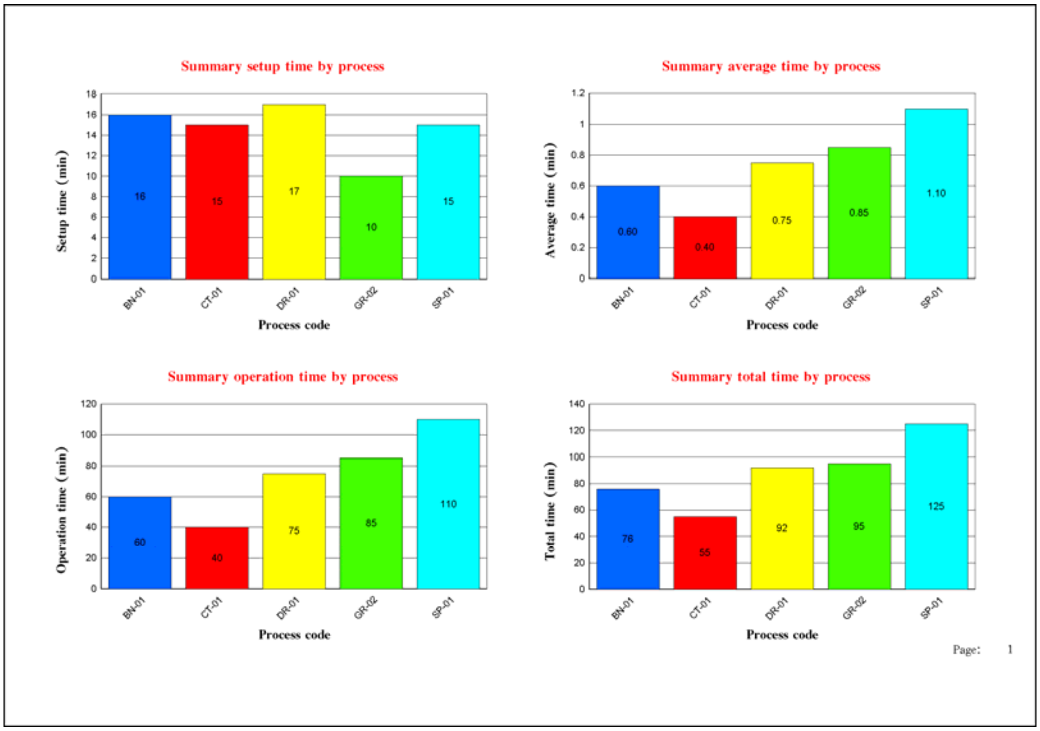
Summary: Raw Material Cost: 47,042.67 baht Setup cost: 203.34 baht Total cost: 60,861.96 baht

Purchased Part Cost: 10,887.60 baht Operation cost: 2,728.34 baht Average Cost: 608.62 baht/unit

Database Application for Production Planning Page: 2

Production reports

Summary chart of time estimation represented by selected process – Page 1



Summary chart of time estimation represented by selected process – Page 2

Report of manufacturing time – Represented by process

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Part Code: CD-003/01 Part Name: Left-Side Plate

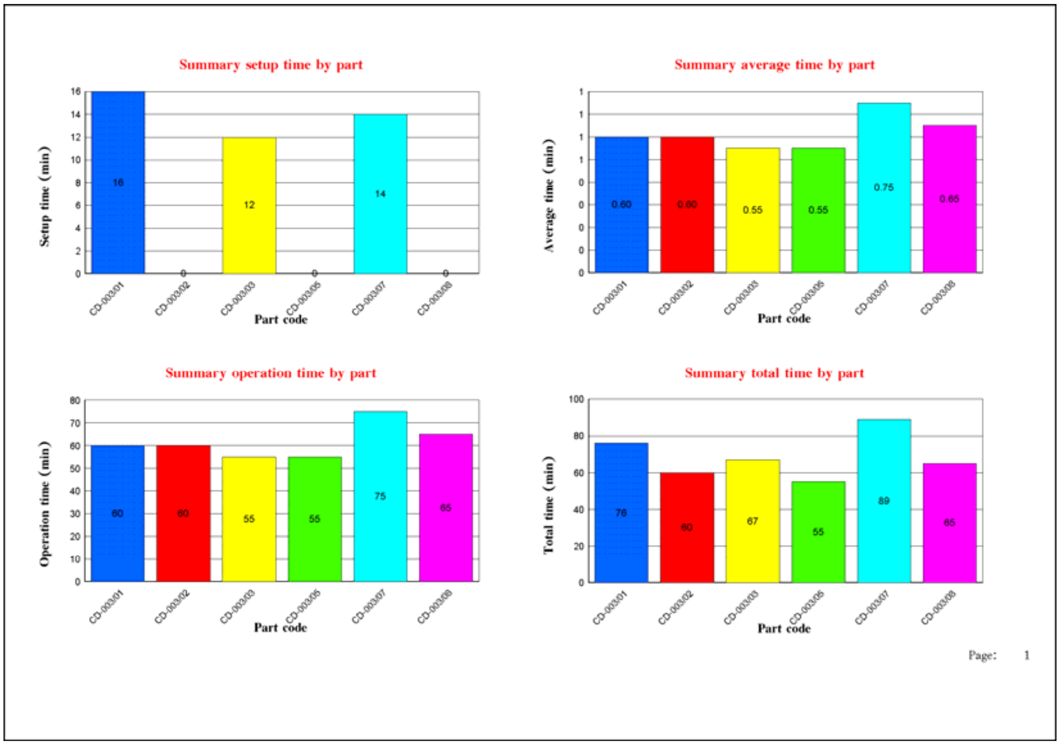
Process code	Process name	Quantity	Manpower	Setup time	Average time	Operation time	Total time
BN-01	Straight-Banding	100	2	16	0.60	60	76
CT-01	Cutting	100	3	15	0.40	40	55
DR-01	Drilling	100	1	17	0.75	75	92
GR-02	Through Grooving	100	2	10	0.85	85	95
SP-01	Finishing	100	3	15	1.10	110	125

Summary: Setup time: 73 min(s) Average time: 3.70 min(s)/piece

Operation time: 370 min(s) Total time: 443 min(s)

Database Application for Production Planning Page: 2

Summary chart of time estimation represented by selected part – Page 1



Summary chart of time estimation represented by selected part – Page 2

Report of manufacturing time – Represented by part

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Process Code: BN-01 Process Name: Straight-Banding

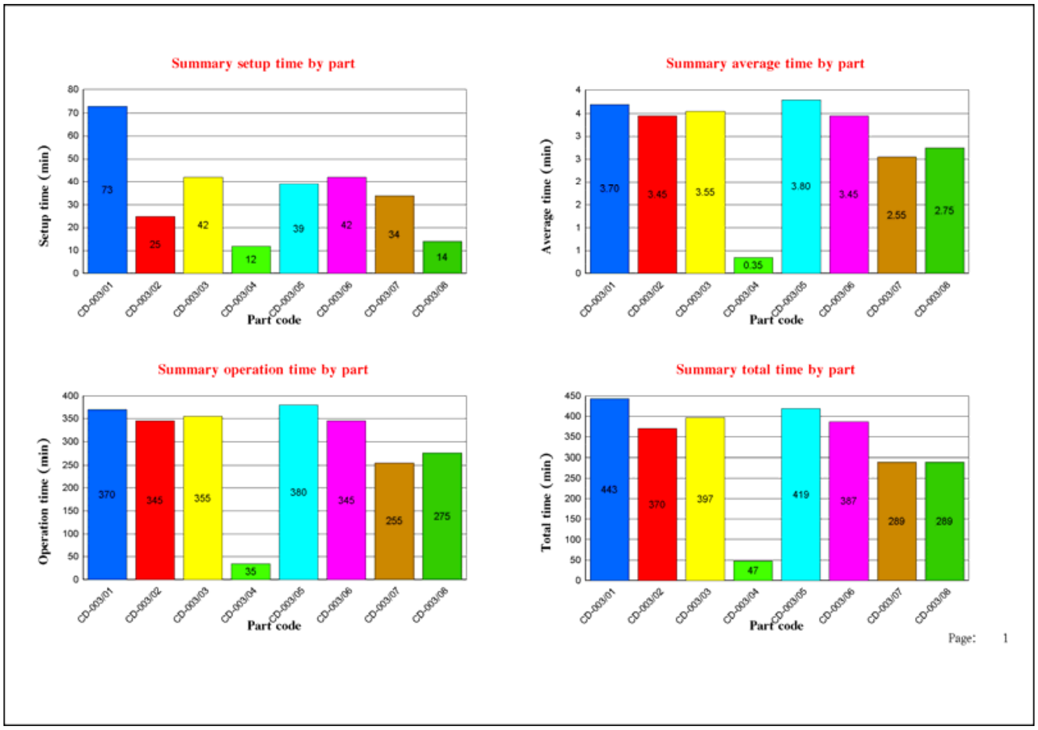
Part Code	Part Name	Quantity	Manpower	Setup time	Average time	Operation time	Total time
CD-003/01	Left-Side Plate	100	2	16	0.60	60	76
CD-003/02	Right-Side Plate	100	2	0	0.60	60	60
CD-003/03	Vertical Fixed Plate	100	2	12	0.55	55	67
CD-003/05	Top Plate	100	2	0	0.55	55	55
CD-003/07	Bottom Plate	100	2	14	0.75	75	89
CD-003/08	Base Plate	100	2	0	0.65	65	65

Summary: Setup time: 42.00 min(s) Average time: 3.70 min(s)/piece

Operation time: 370.00 min(s) Total time: 412.00 min(s)

Database Application for Production Planning Page: 2

Summary chart of time estimation represented by part – Page 1



Summary chart of time estimation represented by part – Page 2

Summary report of manufacturing time – Represented by part

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Part Code	Part Name	Quantity	Setup time	Average time	Operation time	Total time
CD-003/01	Left-Side Plate	100	73	3.70	370	443
CD-003/02	Right-Side Plate	100	25	3.45	345	370
CD-003/03	Vertical Fixed Plate	100	42	3.55	355	397
CD-003/04	Back Plate	100	12	0.35	35	47
CD-003/05	Top Plate	100	39	3.80	380	419
CD-003/06	Horizontal Fixed Plate	100	42	3.45	345	387
CD-003/07	Bottom Plate	100	34	2.55	255	289
CD-003/08	Base Plate	100	14	2.75	275	289

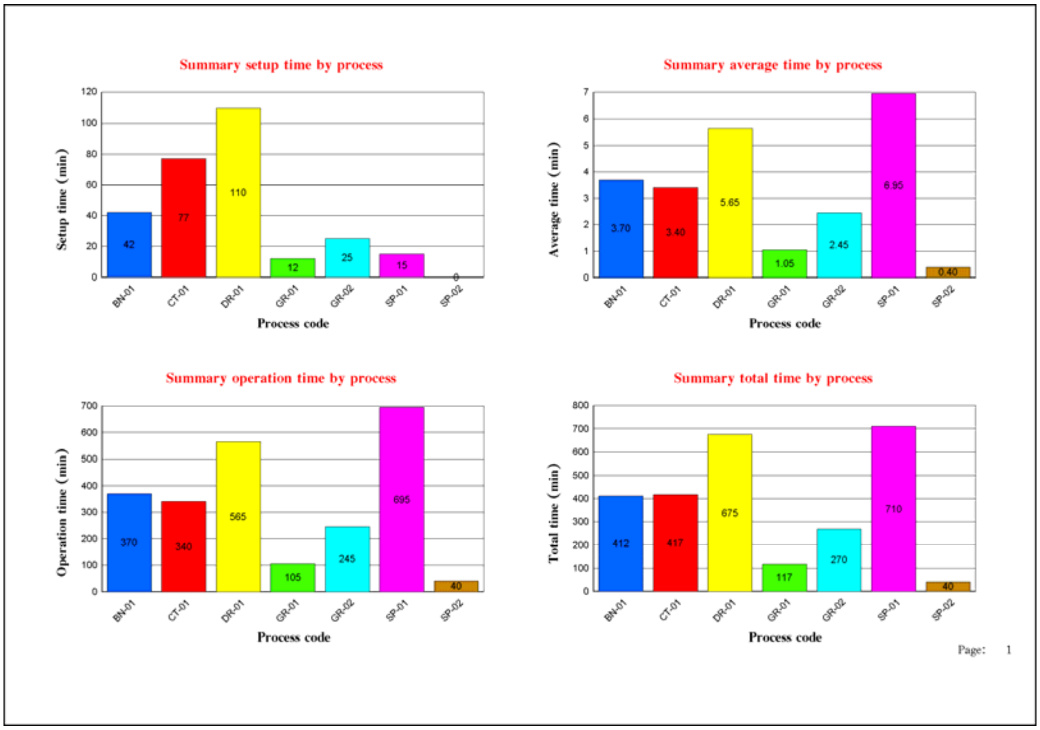
Summary:

Setup time: 281 min(s) Average time: 24 min(s)/piece

Operation time: 2,360 min(s) Total time: 2,641 min(s)

Database Application for Production Planning Page: 2

Summary chart of time estimation represented by process – Page 1



Summary chart of time estimation represented by process – Page 2

G-SCOP **Summary report of manufacturing time – Represented by process**

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Process Code	Process Name	Quantity	Setup time	Average time	Operationt time	Total time
BN-01	Straight-Banding	100	42	3.70	370	412
CT-01	Cutting	100	77	3.40	340	417
DR-01	Drilling	100	110	5.65	565	675
GR-01	Distant Grooving	100	12	1.05	105	117
GR-02	Through Grooving	100	25	2.45	245	270
SP-01	Finishing	100	15	6.95	695	710
SP-02	Pin-up	100	0	0.40	40	40

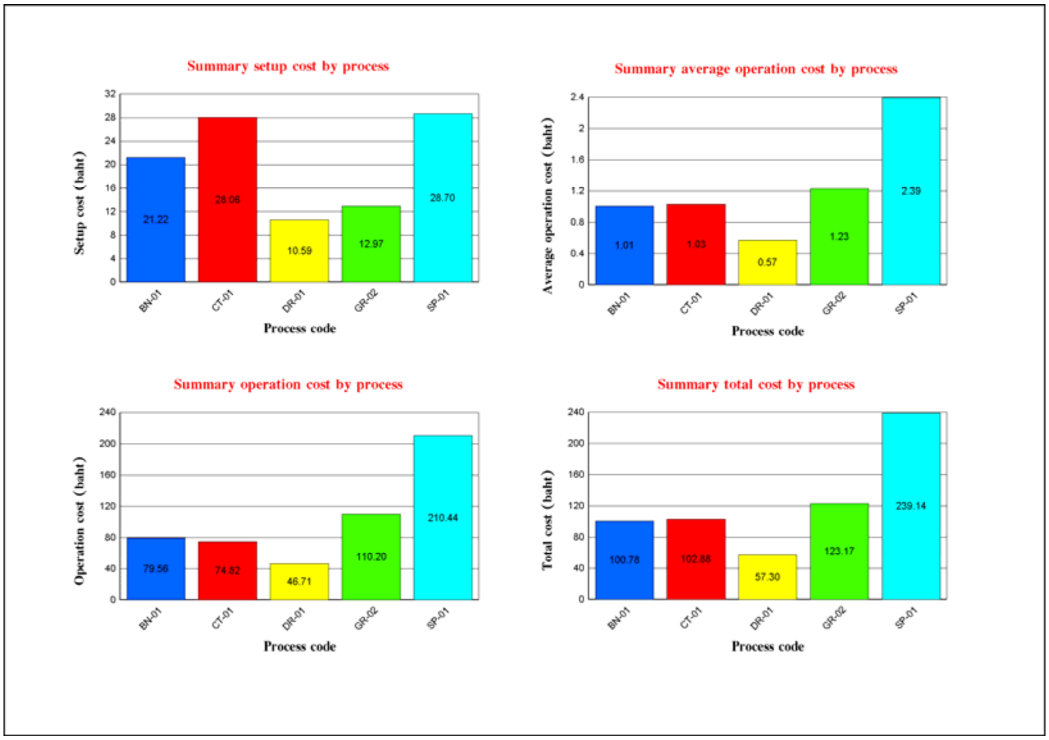
Summary: Setup time: 281 min(s) Average time: 24 min(s)/piece

Operation time: 2,360 min(s) Total time: 2,641 min(s)

Database Application for Production Planning Page: 2

Reports of manufacturing cost

Summary chart of cost estimation represented by selected process – Page 1



Summary of cost estimation represented by selected process – Page 2

G-SCOP

Report of manufacturing cost– Represented by process

Print Date: 27 juillet 2007

Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

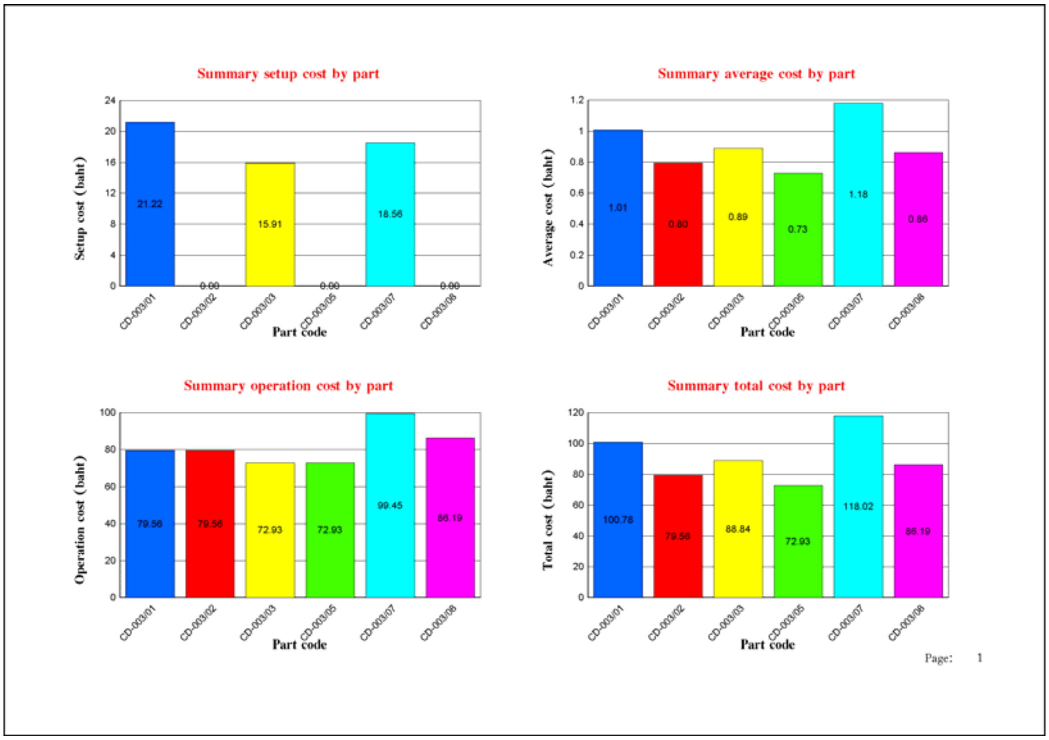
Part Code: CD-003/01 Part Name: Left-Side Plate

Process code	Process name	Quantity	Man	Average hire rate	Setup cost	Operation cost	Total cost	Cost per unit
BN-01	Straight-Banding	100	2	318.25	21.22	79.56	100.78	1.008
CT-01	Cutting	100	3	299.29	28.06	74.82	102.88	1.029
DR-01	Drilling	100	1	298.95	10.59	46.71	57.30	.573
GR-02	Through Grooving	100	2	311.17	12.97	110.20	123.17	1.232
SP-01	Finishing	100	3	306.10	28.70	210.44	239.14	2.391

Summary: Setup cost: **101.53** baht Operation cost: **521.74** baht Total cost: **623.27** baht Cost per unit: **6.23** baht/unit

Database Application for Production Planning Page: 2

Summary chart of cost estimation represented by selected part – Page 1



Summary of cost estimation represented by selected part – Page 2

Report of manufacturing cost– Represented by part

Print Date: 27 juillet 2007

Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

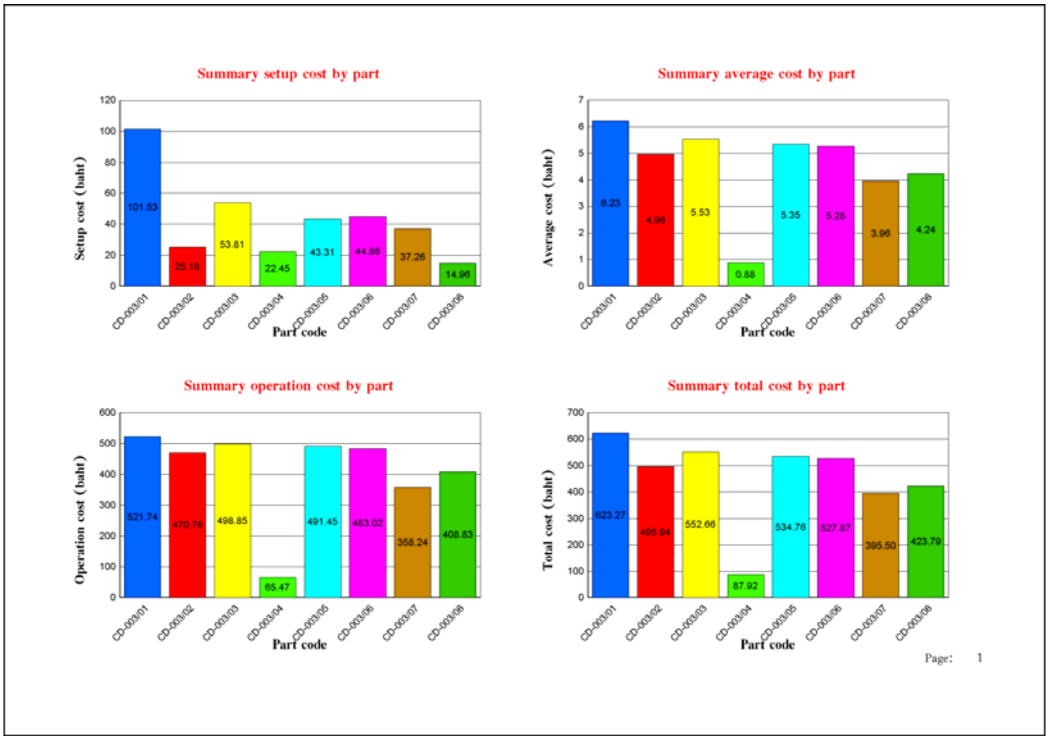
Process Code: BN-01 Process Name: Straight-Banding

Part Code	Part Name	Quantity	Average hire rate	Setup cost	Operation cost	Total cost	Cost per unit
CD-003/01	Left-Side Plate	100	2	21.22	79.56	100.78	1.008
CD-003/02	Right-Side Plate	100	2	0.00	79.56	79.56	.796
CD-003/03	Vertical Fixed Plate	100	2	15.91	72.93	88.84	.888
CD-003/05	Top Plate	100	2	0.00	72.93	72.93	.729
CD-003/07	Bottom Plate	100	2	18.56	99.45	118.02	1.180
CD-003/08	Base Plate	100	2	0.00	86.19	86.19	.862

Summary: Setup cost: **55.69** baht Operation cost: **490.64** baht Total cost: **546.33** baht Cost per unit: **5.46** baht/unit

Database Application for Production Planning Page: 2

Summary chart of cost estimation represented by part – Page 1



Summary of cost estimation represented by part – Page 2

Summary report of manufacturing cost – Represented by part

Print Date: 27 juillet 2007

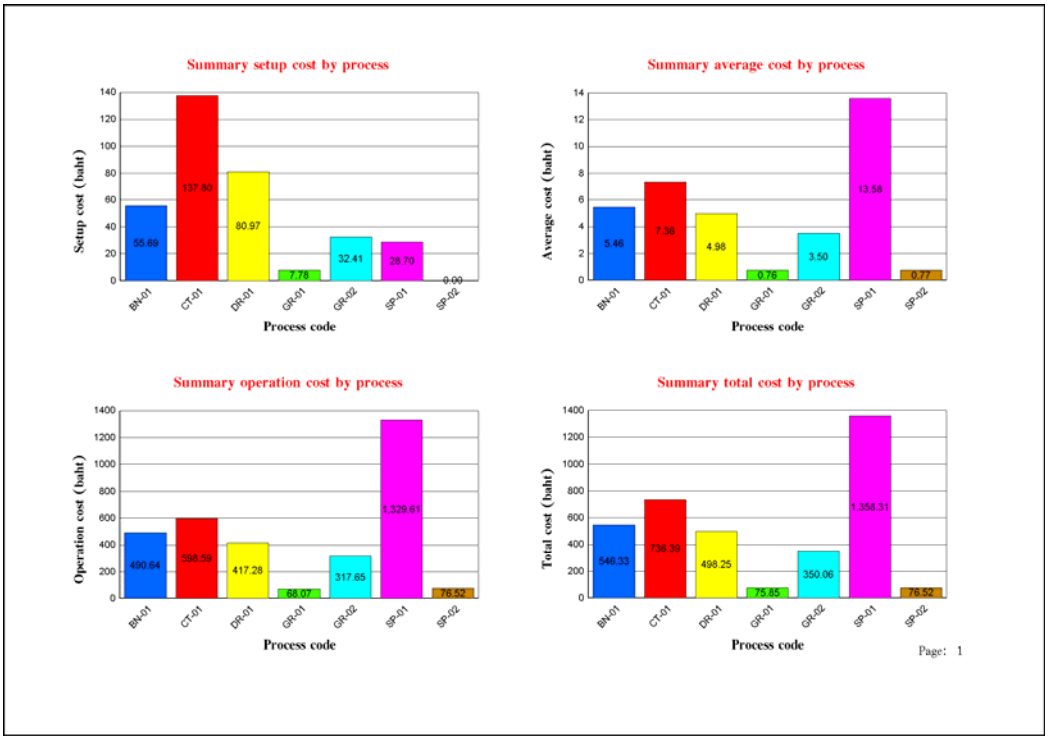
Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Part Code	Part Name	Quantity	Manpower	Setup cost	Operation cost	Total cost	Cost per unit
CD-003/01	Left-Side Plate	100	2	101.53	521.74	623.27	6.233
CD-003/02	Right-Side Plate	100	2	25.18	470.76	495.94	4.959
CD-003/03	Vertical Fixed Plate	100	2	53.81	498.85	552.66	5.527
CD-003/04	Back Plate	100	3	22.45	65.47	87.92	.879
CD-003/05	Top Plate	100	2	43.31	491.45	534.76	5.348
CD-003/06	Horizontal Fixed Plate	100	2	44.86	483.02	527.87	5.279
CD-003/07	Bottom Plate	100	2	37.26	358.24	395.50	3.955
CD-003/08	Base Plate	100	2	14.96	408.83	423.79	4.238

Summary: Setup cost: **343.35** baht Operation cost: **3,298.36** baht Total cost: **3,641.71** baht Cost per unit: **36.42** baht/unit

Database Application for Production Planning Page: 2

Summary chart of cost estimation represented by process – Page 1



Summary of cost estimation represented by process – Page 2

G-SCOP

Summary report of manufacturing cost – Represented by process

Print Date: 27 juillet 2007

Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Process Code	Process Name	Quantity	Manpower	Average hire rate	Setup cost	Operation cost	Total cost	Cost per unit
BN-01	Straight-Banding	100	2	318.25	55.69	490.64	546.33	5.463
CT-01	Cutting	100	3	299.29	137.80	598.59	736.39	7.364
DR-01	Drilling	100	1	298.95	80.97	417.28	498.25	4.983
GR-01	Distant Grooving	100	1	311.17	7.78	68.07	75.85	.758
GR-02	Through Grooving	100	2	311.17	32.41	317.65	350.06	3.501
SP-01	Finishing	100	3	306.10	28.70	1,329.61	1,358.31	13.583
SP-02	Pin-up	100	3	306.10	0.00	76.52	76.52	.765

Summary: Setup cost: **343.35** baht Operation cost: **3,298.36** baht Total cost: **3,641.71** baht Cost per unit: **36.42** baht/unit

Database Application for Production Planning Page: 2

Additional reports:

Summary report of setup time estimation

Summary of Time Estimation: Setup Time **Product:** **DC100**

Part Code	Part Name	Cutting	Distant Grooving	Drilling	Finishing	Straight-Banding	Through Grooving	Total
DC100-01	Left-Side Plate	10	10	15		10		45
DC100-02	Right-Side Plate	10	10	15		10		45
DC100-03	Vertical-Center Plate	10		15		10		35
DC100-04	Back-Upper Plate	10						10
DC100-05	Back-Lower Plate	10		15		10		35
DC100-06	Upper-Fixed Plate	10		15		10		35
DC100-07	Lower-Fixed Plate	10		15		10		35
DC100-08	Horizontal-Center Plate	10		15		10		35
DC100-09	Top Plate	10		15		10	10	45
Total		90	20	120		80	10	320

Database Application for Production Planning

Summary report of operation time estimation

Summary Report of Operation Time

Print Date: 27 juillet 2007

Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Part Code	Part Name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Straight-Banding	Through Grooving
CD-003/01	Left-Side Plate	100	40		75	110		60	85
CD-003/02	Right-Side Plate	100	35		80	90		60	80
CD-003/03	Vertical Fixed Plate	100	45		75	100		55	80
CD-003/04	Back Plate	100	35						
CD-003/05	Top Plate	100	45	105	70	105		55	
CD-003/06	Horizontal Fixed Plate	100	60		140	105	40		
CD-003/07	Bottom Plate	100	40		65	75		75	
CD-003/08	Base Plate	100	40		60	110		65	
Total			**340**	**105**	**565**	**695**	**40**	**370**	**245**

Database Application for Production Planning

Summary report of man-hours

Summary Report of Man-hours

Print Date: 27 juillet 2007

Production order no.: CN-001 **Product:** CD-003 **Product Name:** CD Cabinet Quantity: 100 Units

Part Code	Part Name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Straight-Banding	Through Grooving
CD-003/01	Left-Side Plate	100	120		75	330		120	170
CD-003/02	Right-Side Plate	100	105		80	270		120	160
CD-003/03	Vertical Fixed Plate	100	135		75	300		110	160
CD-003/04	Back Plate	100	105						
CD-003/05	Top Plate	100	135	105	105	315		110	
CD-003/06	Horizontal Fixed Plate	100	120		210	315	120		
CD-003/07	Bottom Plate	100	120		65	225		150	
CD-003/08	Base Plate	100	120		60	330		130	
Total			**960**	**105**	**670**	**2,085**	**120**	**740**	**490**

* Man-hours = Operation time x Manpower

Database Application for Production Planning

Summary report of manpower

G-SCOP

Summary Report of Manpower

Print Date: 27 juillet 2007

Production order no.: CN-001 Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Part Code	Part Name	Quantity	Cutting	Distant Grooving	Drilling	Finishing	Pin-up	Straight-Banding	Through Grooving
CD-003/01	Left-Side Plate	100	3		1	3		2	2
CD-003/02	Right-Side Plate	100	3		1	3		2	2
CD-003/03	Vertical Fixed Plate	100	3		1	3		2	2
CD-003/04	Back Plate	100	3						
CD-003/05	Top Plate	100	3	1	1	3		2	
CD-003/06	Horizontal Fixed Plate	100	2		2	3	3		
CD-003/07	Bottom Plate	100	3		1	3		2	
CD-003/08	Base Plate	100	3		1	3		2	

Database Application for Production Planning

Report defects and reworks

Report of defects and reworks – represented by part

G-SCOP SCIENCES POUR LA CONCEPTION, L'OPTIMISATION ET LA PRODUCTION

Report of Defects, Delays, and Reworks – Represented by part

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Process Code: DR-01 Process Name: Drilling

Part Code	Part Name	Quantity	Good	Defect	Rework	Error	Percentage
CD-003/01	Left-Side Plate	100	100	0	0	0	0.00
CD-003/02	Right-Side Plate	100	100	0	0	0	0.00
CD-003/03	Vertical Fixed Plate	100	100	0	0	0	0.00
CD-003/05	Top Plate	100	100	0	0	0	0.00
CD-003/06	Horizontal Fixed Plate	200	199	0	1	1	0.50
CD-003/07	Bottom Plate	100	100	0	0	0	0.00
CD-003/08	Base Plate	100	100	0	0	0	0.00

Summary:	Total:	800	Piece(s)	Defect:	0.00	Piece(s)	Percentage:	0.00	%
	Good:	799	Piece(s)	Rework:	1.00	Piece(s)	Percentage:	0.13	%
				Total:	1.00	Piece(s)	Percentage:	0.13	%

Database Application for Production Planning Page: 1

Report of defects and reworks – represented by process

G-SCOP

Report of Defects, Delays, and Reworks – Represented by process

Print date: 27 juillet 2007

Production order no.: CN-001

Product: CD-003 Product Name: CD Cabinet Quantity: 100 Units

Process Code: BN-01 Process Name: Straight-Banding

Process Code	Process Name	Quantity	Good	Defect	Rework	Error	Percentage
BN-01	Straight-Banding	100	100	0	0	0	0.00
CT-01	Cutting	100	100	0	0	0	0.00
DR-01	Drilling	100	100	0	0	0	0.00
GR-01	Distant Grooving	100	98	1	1	2	2.00
SP-01	Finishing	100	98	2	0	2	2.00

Summary:									
	Total:	500	Pieces(s)	Defect:	3	Pieces(s)	Percentage:	1	%
	Good:	496	Pieces(s)	Rework:	1	Pieces(s)	Percentage:	0	%
				Total:	4	Pieces(s)	Percentage:	1	%

Database Application for Production Planning Page: 1

Conception intégrée de meubles réalisés en panneaux de fibres ou de particules

Résumé

Le contexte de globalisation et la volonté de mettre au plus vite sur le marché les produits ou services obligent les entreprises à intégrer des unités délocalisés dès la phase de conception. Cette intégration de domaines très différents rend de plus en plus complexe le processus de conception. Les différents acteurs doivent alors pour collaborer introduire leurs propres contraintes en juste besoin. Ils doivent pour cela pouvoir travailler sur leurs propres vues dans un système multi-acteur mais aussi multidisciplinaire. Nous montrons dans cette étude comment le système de conception collaborative proposé permet de résoudre les problèmes de complexité imaginaire et facilite l'approche de problèmes de complexité réelle.

Le développement d'un système collaboratif de conception de meubles réalisés en panneaux de fibres ou de particules permet de manière pragmatique de valider notre approche. Ce système intègre un designer, ou styliste, proposant en esquisse les formes et les principales dimensions d'un meuble répondant à un cahier des charges. Un technologue, un spécialiste des assemblages, un mécanicien et un homme de production permettent progressivement de réaliser les choix technologiques adéquats, de dimensionner les différents éléments en tenant compte de critères de qualité et d'évaluer le coût final d'obtention du produit.

Mots-clés

Conception intégrée, Complexité, Optimisation, Evaluation du coût, Fabrication, Meuble en panneaux de particules ou fibres

Abstract

The globalization and the condition of the lead time to market bring the companies into a high competitive environment and lead the complexity into the design process. The companies are obliged to integrate the delocalized units and the design actors from different disciplines in order to work as a multidisciplinary design team. The design actors have to introduce their own constraints and information in the notion of just need. They must also be able to work on their own views base on the multi-actor system. We propose in this study an integrated design system that supports the design team to integrate knowledge from different design actors in different disciplines and also to solve the problem of imaginary complexity..

The development of this design system for the wood furniture made of particleboard and medium-density fiberboard permits us to validate our proposition in a pragmatic way. This system integrates different actors into the design process. A designer propose the global form and shape of the product, regarding to the specifications, into a conceptual product model; a technologist transforms the conceptual product model into the design system; an assembler, a mechanician, and a manufacturer realize progressively the technological choices, calculate the dimensions of various elements by taking into account the design in term of cost and quality, and evaluate the cost of the final design.

Keywords

Integrated design, Complexity, Optimization, Cost evaluation, Manufacturing, Furniture made of particleboard and fiberboard