# PRODUCT AND PROCESS INFORMATION INTERACTIONS IN ASSEMBLY DECISION SUPPORT SYSTEMS

By

Jesús Manuel Dorador González

**A Doctoral Thesis**

Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University

June 2001

# DEDICATION

To Soco with all my love,
You are the reason of my happiness.

To my mother, your example is the light in my life.
To my father, thanks for giving always the best of yourself.
To Luis & Mónica, Nunú, Tomás & Tomy, Francisco, Juan Pablo,
thanks for all your support and love.

To Aita, who always told me "Look how nice is to study"

To Our Lady of Guadalupe, who was present with me all this time.
To God, thanks for this opportunity.

## ACKNOWLEDGEMENTS

This work was realised at Loughborough University with the financial support of the Universidad Nacional Autónoma de México, from the CONACYT, México, and from The Foreign and Commonwealth Office of the United Kingdom through a British Chevening Scholarship.

First of all, a special thanks to Dr. Robert Young, for guiding my work with professionalism and enthusiasm.

A very special thank you for Dr. Keith Toh and Dr. Carlos Costa, who were always willing to help and advise, friends like you are very difficult to find.

Thanks to all my friends at the lab, you made it a second home. Every one of you left something special in my life.

Special thanks to all the people in Alstom Electrical Machines, that provided lots of information and support to my research.

Thanks to all my friends that kept in touch through this time shortening the distance between us.

# ABSTRACT

A characteristic of concurrent engineering is the intensive information interchange between areas that are involved through the product life cycle. Shared information structures to integrate different software applications have become necessary to support effectively the interchange of information. While much work has been done into the concepts of Product and Manufacturing Models, there is a need to make them able to support Assembly related activities.

The research reported in this thesis explores and defines the structures of a Product Model and a Manufacturing Model to support assembly related information. These information models support the product development process, especially during the early stages of the product life cycle. The structures defined for the models allow information interactions between them and with application software; these interactions are essential to support an effective concurrent environment. The Product Model is a source and repository of the product information, whilst the Manufacturing Model holds information about the manufacturing processes and resources of an enterprise. A combination of methods was proposed in order to define the structure for the information models.

An experimental software system was created and used to show that the structure defined for the Product Model and the Manufacturing Model can support a range of assembly-related software applications through the concurrent development of the product, system and process, from conceptual design through to planning. The applications implemented in the experimental system were Design for Assembly and Assembly Process Planning. The real data used for the tests was obtained from an industrial collaborator who manufactures large electrical machines.

This research contributes to the understanding of the general structural requirements of the decision support systems based on information models, and to the integration of Design for Assembly and Assembly Process Planning.

# TABLE OF CONTENTS

# GLOSSARY OF TERMS

| | |
|---|---|
| **Assembly** | 1. A combination of parts and possibly raw materials put together to make up a composite article. (BSI, 1975)<br>2. A part which is made up of two or more parts i.e. a group of components, raw material and sub–assemblies which are put together to make up a composite article. (Ellis, 1993) |
| **Assembly Plan** | An assembly plan describes how to assemble the product. (Cunha, *et al.*, 1999). |
| **Assembly Process Planning (APP)** | Process Planning performed for Assembly. (See Process Planning) |
| **Assembly Production Planning and Control** | Assembly Production Planning and Control systems are responsible for planning the utilisation of production resources which are required, over a planning horizon, considering some demand pattern (Cunha *et al.*, 1999). |
| **Bill Of Materials** | A list of all parts, subassemblies and raw materials that constitute a particular assembly, showing the quantity of each required. (BSI, 1975) |
| **Component** | A uniquely identifiable product that is considered indivisible for a particular planning or control purpose. (BSI, 1975) |
| **Concurrent Engineering** | "The systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause developers, from the outset, to consider all elements of the product life cycle from concept through disposal, including quality, cost, schedule and user requirements" (Belson, 1994). |

| | |
|---|---|
| **Design For Assembly (DFA)** | Design For Assembly (DFA) set of techniques applied to reduce the cost of the assembly by optimising the design of the product. |
| **Information** | 1. Data that has been interpreted to make it meaningful. (Ellis, 1993)<br>2. The meaning that a human assigns to data by means of the conventions used in its representation. (CIRP, 1990) |
| **Information Models** | A model of information showing relationships between items. (Ellis, 1993) – see model |
| **Integration Environment** | The Integration Environment shown linking the Information Models and the data-driven applications is required to enable these elements to work together, even if them are based on different platforms or are located in different sites. This integration environment has to satisfy the requirements of the different elements, so that the models may store and maintain the information and the applications perform particular functions and access the information (Harding, 1996) |
| **Knowledge** | An understanding of how information may be utilised. (Ellis, 1993) |
| **Manufacturing Model** | The term Manufacturing Model is used to define, describe and capture the manufacturing situation of a particular enterprise (Young and Bell, 1995), (Molina, 1995) |
| **Model** | A model is a representation or description of an entity or a system, describing only the aspects considered to be relevant in the context of its purpose. (International Standards Organisation, 1997), |
| **Parent Assembly** | Refers to a complete product |
| **Process Planning** | The activity which determines the operations required, and their sequence to produce a part or assembly (BSI, 1975) |

| | |
|---|---|
| **Product** | A part or service which is sold by an organisation. (Ellis, 1993) |
| **Product - Original** | Products designed according to customer specifications. |
| **Product - Standard** | The Standard product is referred to those parts, subassemblies of parent assemblies that are commercially available, such as screws, nuts, bolts, and bearings. |
| **Product - Variant** | A Variant Product is defined as a product that is part of a family of products, sharing characteristics of the family of products. |
| **Product Model** | "A Product Model is a computer readable representation of all product related data" (Young and Bell, 1995) |
| **Subassembly** | 1. An assembly which is used at a higher level to make up another assembly (CIRP, 1990)<br>2. Any assembly which is not a final assembly. (BSI, 1975) |

# 1. INTRODUCTION

## 1.1. Context

Global market pressures, international competition, shorter product lifecycles, increasing product diversity, growing demands from the customers for better products and shorter delivery times are some of the factors that affect industrial production. In order to solve those issues, technological developments such as information technology, new design strategies, new processing techniques, and flexible production systems play a key role. In order to achieve significant improvements in the product cost, functionality and time to market, changes have to be done in the most influential stage of the product, that is, in the early stages of the design. Several industries and researchers have proved that working in a concurrent engineering environment help to perform those changes.

A characteristic of concurrent engineering is the intensive information interchange between the different areas that are involved through the product life cycle. This information includes an extensive share of product and process data to allow the applications to work concurrently. Significant advances in the computational power of desktop computers have allowed the use of software applications to support activities through the product life cycle. In order to avoid errors and reduce the time in the translation of the information from one system to another, a shared information structure approach is necessary. This approach based on information models is a major feature of this research.

Assembly is a very important process in production systems, almost all the products consist of several parts and subassemblies that have been manufactured at different times with different machines and frequently in different factory locations. Furthermore, assembly represents a fundamental factor in the economy of the enterprise. According to published surveys, assembly accounts for over 50% of total production time and for 20% of the total unit production cost.

In spite of their importance and effect on the cost of a product, consideration for assembly-related activities is given mostly in the final stages of design. Rationalisation of assembly involves the improvement of the quality of assembled products and their cost. There are two main ways for performing this rationalisation; one is using Design for Assembly (DFA) methods, which provide the designer with tools for evaluating the assemblability of products in the detailed design stage of the product design cycle. The second way is by applying assembly representation methods, which focus on CAD representation models and Assembly Process Planning (APP).

Commercial systems that aid DFA evaluation are available, such as DFMA© and TeamSET©. The first is the best known software for performing DFA analysis and the second contains a module that uses the approach known as the Lucas method. Although the two applications are very useful for design evaluation, they were not designed for readily sharing information with other software systems in a concurrent engineering environment. Furthermore, their strength is in the detailed design stage of the product lifecycle and do not provide help to the designer at the early stages of design.

In order to provide support to team-based concurrent engineering, it is important that the tools developed are able to operate in an integrated environment. This is not the case of a large number of the software tools currently used in industry for Process Planning.

A generally accepted way of providing an integrated environment is the utilisation of Information Models. Much work has been done into the concepts of the Product Model and the Manufacturing Model. In the first, the product with its characteristics is captured, and in the second, manufacturing information is described and stored.

The work reported on this thesis contributes to the area of decision support by defining information model structures to support assembly-related activities in the product life cycle, focusing on DFA and APP application during the early stages of product design.

The structures of the Product and Manufacturing Models have been defined to achieve a suitable representation of information that intensifies the interactions between models and between applications. Emphasis has been placed on investigating those relationships in order to support assembly processes and strategies, due to the dependency existing with the product.

## 1.2. Aim and Objectives

The aim of this research is to make a significant contribution by identifying the information related to assembly, which could be used in the design and manufacture stages to reduce the product development cycle. The necessary structures are defined to support the interactions between the Product and Manufacturing Models and to allow the information interactions between Design for Assembly and Assembly Planning activities.

The primary objectives of the research are:

1. Define a Manufacturing Model capable of supporting assembly process information and define the way in which that information will be captured.
2. Define a Product Model structure that allows interchange of information with the Manufacturing Model and with application software.
3. Explore how these information models support the interaction between Design for Assembly and Assembly Planning

The secondary objectives of the research are:

4. Critically study the state of the art of modelling information of assembly processes, Design for Assembly and Assembly Process Planning in order to identify useful approaches for providing information support in a Concurrent Engineering environment.
5. Explore the information requirements imposed by assembly-related activities through the product life cycle.
6. Design a testing software system capable of integrating Design for Assembly and Assembly Process Planning activities by using, retrieving and storing information of the Product Model and the Manufacturing Model.
7. Populate the testing system with real data obtained from companies collaborating in the Manufacturing Information Models (MIM) project in order to perform case studies for testing and validate the proposed ideas.

## 1.3. Structure of the Thesis

The present section sets the work into context for readers.

Chapter 2 presents a literature review relating to the relevant areas of research, it starts exploring information modelling in decision support systems, with reference to the Product Model and the Manufacturing Model. Then a review of the current research on assembly-related information in the product and process design cycle is presented.

Chapter 3 provides a description of the research environment in which this work was carried out. The combination of methodologies selected for the development of the information models is presented, emphasising on the benefits that the use of IDEF0, IDEF3 and UML provided to the work.

Chapter 4 raises the issues related to product and process information interactions in assembly decision support systems, highlighting the author's opinion of the feasibility of supporting those interactions through the use of information models. This chapter also defines the contents of the following chapters, which provide a description and understanding of the proposed information models and their interactions. A description of the DFA and APP applications and their interactions through the use of the information models is given.

Chapter 5 describes the process followed for defining the structure of the information models.

Chapter 6 evaluates DFA and APP through the concurrent development cycle of the product, system and process in order to identify their information requirements.

Chapter 7 presents the structures defined for the Product and Manufacturing Models, and explains the way in which interactions between them take place.

Chapter 8 presents the implementation of the testing environment and describes the experiments performed to test the ideas.

Chapter 9 discusses the results of the tests done with the use of information provided by the collaborators of the M1M project

Chapter 10 presents a discussion about the main topics of the research, the conclusions of this research, as well as the recommendations for further developments in this area.

# 2. LITERATURE REVIEW

## 2.1. Introduction

This chapter presents the results of the literature survey performed by the author about the topics related to this research. Section 2.2 presents a general background about Concurrent Engineering and the importance of assembly activities in the product development cycle.

Section 2.3 introduces the trends about information models in decision supports systems, explaining the concepts of Product Model and Manufacturing Model.

Section 2.4 reviews issues related to assembly, including Design For Assembly, Assembly Process Planning, Assembly Modelling, and the inclusion of assembly information into the Product and Manufacturing Models.

## 2.2. Background

Nowadays industrial production is under great pressure. The global market pressure for international competition, shortest life of the products, increasing product diversity, and major demands of the clients on quality and shorter delivering times are only some of the factors that are affecting the way of producing in the modern enterprise.

To solve those problems the technological developments play a key role, offering to the enterprises new opportunities to optimise price, quality and delivery time. The technological developments are, among other things: information technology, new design strategies, new materials, quality improvement techniques, new processing techniques and the availability of flexible production systems. Using those technologies, the enterprises can apply advanced production systems, achieve an integral approach in the materials and information flows, have tighter quality control systems and improve the working conditions (Rampersad, 1995a).

In the 1980's the idea that the complete automation of systems would solve all the problems was very common. Predictions about the future told that in the middle 1990's the CIM systems would be a reality in the enterprises and for the year 2000 complete automation would be possible. It is true that some enterprises have achieved that level of automation, but the cost has been enormous. The common enterprises are very far from achieving those automation levels, but they have to survive in this highly competitive world and for doing so they have to improve their way of producing.

Assembly is a very important process in the production systems, almost all the products consist of several parts and subassemblies that have been manufactured at different times with different machines, and some times in different places.

Furthermore, assembly represents a fundamental factor in the economy of the enterprise. According to various published surveys, assembly accounts for over 50% of total production time and for 20% of the total unit production cost. Of that 20%, slightly less than half was attributable to intermediate and final assembly operations, while the other part was attributable to set-up and other assembly support functions (Nevins and Whitney, 1978), (Martin-Vega and Brown, 1995), (Mo, *et al.*, 1999). In the automotive industry, 50% of the direct labour costs are in the area of assembly and in precision instruments is between 20 and 70%. Approximately one third of the total workforce is engaged in assembly (Martin-Vega and Brown, 1995) and assembly accounts for approximately 20 to 50 per cent of the total throughput time (Rampersad, 1995a). Assembly can also account for a great deal of hidden cost, such as scrap and rework (Whitney, 1996).

Automation of the assembly operations have been developed slower than other areas, because of the following reasons (Nof, 1997):

1. *Assembly is highly product specific, with highly variable tasks of handling, joining, adjusting, and testing.*

2. *Assembly, as a final production stage, must cope with continuously shifting market requirements in regard to timing, batch sizes and product design or style.*
3. *Assembly automation has, until recently, been difficult to justify economically.*

Despite the importance of assembly activities, they are not commonly evaluated at the early stages of design, and are only used for balancing the production lines. If the manufacturing and assembly activities could be considered at the early stages of the design cycle, the potential of cost saving and design rationalisation would be higher. That is because 80% of the cost associated to a product is defined in the design stage of the product life cycle. Figure 2.1 shows that relation (Belson, 1994), (Syan and Menon, 1994), (Leaney and Wittenberg, 1992), (Fox, 1995), (Andreasen, 1988), (Hsu, 1996).

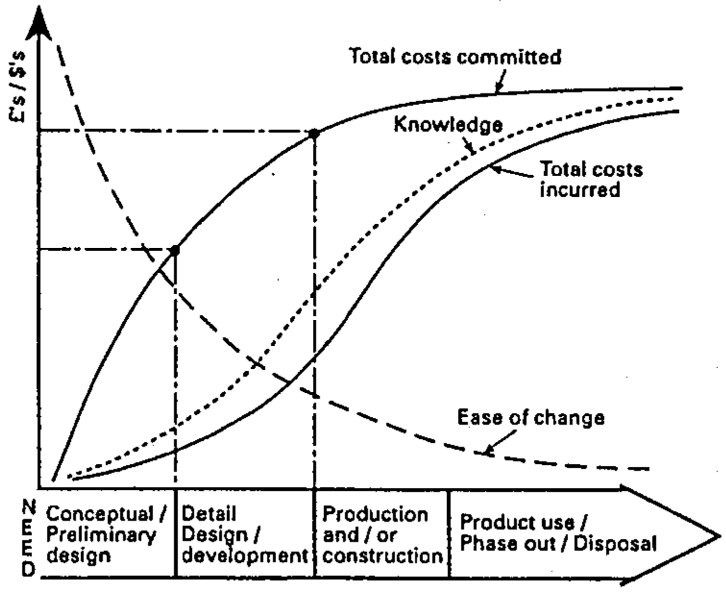



Figure 2.1 Time versus cost of changes (Syan and Menon, 1994)

## 2.3. Information Modelling in Decision Support Systems

A simultaneous way of working, called concurrent engineering, has been established and applied successfully in several industries. In it the different areas that are involved in the product life cycle work together in order to ensure that all the decisions taken during the design process take in account all their consequences. This allows the reduction of delays due to misunderstandings or difference of opinion between areas, problems that are common in the traditional sequential engineering process.

In the concurrent way of working, great amounts of time and resources can be saved, because the changes can be done in the early stages of the product life cycle. In those early stages the changes are cheap and have a great impact in the final cost of the product.

The definition of Concurrent Engineering given by the Institute for Defense Analysis is:

> "The systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause developers, from the outset, to consider all elements of the product life cycle from concept through disposal, including quality, cost, schedule and user requirements" (Belson, 1994).

By applying concurrent engineering, the decisions that are made during the design process can be done with a full understanding of the consequences of these decisions. In order to achieve this, the company must have an appropriate organisational structure that emphasises teamwork and an information system capable of storing and providing information to the design team (Young and Bell, 1995).

One of the main characteristics of Concurrent Engineering is the intensive information interchange (Pels, 1996). This includes sharing information about products and the processes and resources required in their production. Different areas of the enterprise, using a variety of computer software applications, use the same information, so it is convenient to have it centrally stored in information models. The issue of information modelling have been widely recognised and studied by the international research community (Pels, 1996), (Molina, 1995). Other Decision support systems include approaches that use algorithms, agent-based systems, and knowledge-based systems.

### 2.3.1. Information Modelling

A model is a representation of the characteristics of a system. This system could be physical, conceptual or analytical. Commonly the model for representing something physical is a prototype, for something conceptual is a scheme or a method and for something analytical is an equation (Prasad, 1996).

According to the ISO (International Standards Organisation, 1997), a model is a *representation or description of an entity or a system, describing only the aspects considered to be relevant in the context of its purpose.*

The purpose of modelling is to predict the behaviour of the system. As information is one of the most critical aspects in the product life cycle, its modelling is essential to allow the designers to work in a concurrent way.

According to Andreasen (Andreasen, 1994) modelling has the following characteristics: object, property, purpose and user, besides the code used and the medium in which the model is created.

Molina, et al. (Molina, *et al.*, 1994) classified the requirements for information modelling in:

- Capture and represent product information, and manufacturing process and resource information.
- Provide immediate access to information about previous products or process design and transmit design information without loss of intent or detail.
- Offer immediate access to information about manufacturability, reliability, maintainability, safety, performance and other elements of the life cycle.
- Allow access to the most current state of the product or process configuration description as it is developed.
- Keep data to be shared by team members in commonly accessible databases.

### 2.3.2. Product Model

Computer systems that aid the designer during the design cycle stages need information about the product to perform their work. Some of this product information is specific for each software application, but some of it can be shared in order to avoid repetition of the information between systems (Baxter, 1994).

In order to handle the information needed during the product life cycle it is necessary to model it. Product models are used as an aid to the designer in imagining and validating new designs and to communicate these designs to the designer of the production process (Pels, 1996).

All the engineering processes within a business use and create information data. Frequently the same piece of information is duplicated several times. This causes major problems for the management of such data. A shared Product Model with computer tools to provide information for the different views that are required, significantly reduces the number of the required modifications (MOSES, 1994a). The Product Model is the central representation from which all the applications obtain and store data.

An early attempt to integrate the systems was made with the use of protocols for transferring product data between them; the problem was the incompatibility of the systems. Some neutral standards for interchange information as IGES and DXF were developed for solving that problem. With that kind of links between "islands of information" the need for retype the information was avoided, but the problem of the management of information arose. Some Product Data Management (PDM) systems were developed in an attempt to have a system for the interchange of information between systems.

PDM integrates and manages processes, applications and information that define products across multiple systems and media (Philpotts, 1996). PDM systems provide a structure in which all types of information used from design to manufacture, and to end-user support are integrated, stored, managed and controlled (Liu and Xu, 2001).

Those attempts were not the solution to the problem because the information still existed in all the applications and a real share of it was not achieved. Furthermore, the problem of consistency of the information was still there, for example, it is not possible to transform a 2D model into a surface model or a solid model without adding information to the source representation (McKay, 1991).

The reason for the developing of information data models was the introduction of a central database for storing the information instead of using files created by the applications with specific structures for those applications that made very difficult the interchange of information (Pels, 1996). A comparison between interfacing and integration is illustrated in Figure 2.2.

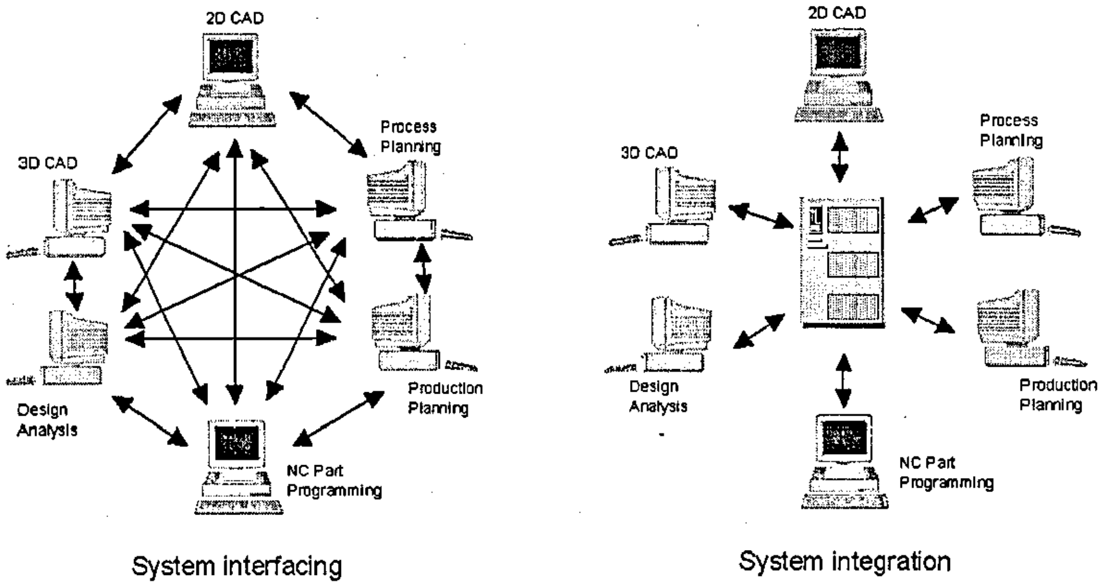



Figure 2.2 Comparison between interfacing and integration (Young and Bell, 1995)

A central database that could hold all the information of the product is known as a 'Product Model'.

A Product Model is a computer representation of product data that contains detailed information about a certain product, so it can support the applications that are interacting in the product's life cycle. The Product Model must allow those applications to have access to the information and also to be a repository for the data created by them. Therefore, the Product Model must be structured in such a way that the applications know where to find and to store information (Baxter, 1994), (MOSES, 1992), (Krause, *et al.*, 1993).

There are several definitions for the Product Model, the one that will be accepted in this document is "*A Product Model is a computer readable representation of all product related data*" (Young and Bell, 1995).

The structure of the Product Model depends on the nature of the product and on the tools used to model the information. The Product Model contents are dependent on the particular product. It is impractical to attempt to build a generic Product Model because the models are bound to contain specific data (Krause *et al.*, 1993), (McKay, 1991).

Krause et al. (Krause *et al.*, 1993) states that product modelling is a broad topic closely related to many other issues in manufacturing engineering and concerns the complete life-cycle, as shown in Figure 2.3. They proposed the division of the Product Model in partial models: Design, Technology Information and Planning Model.

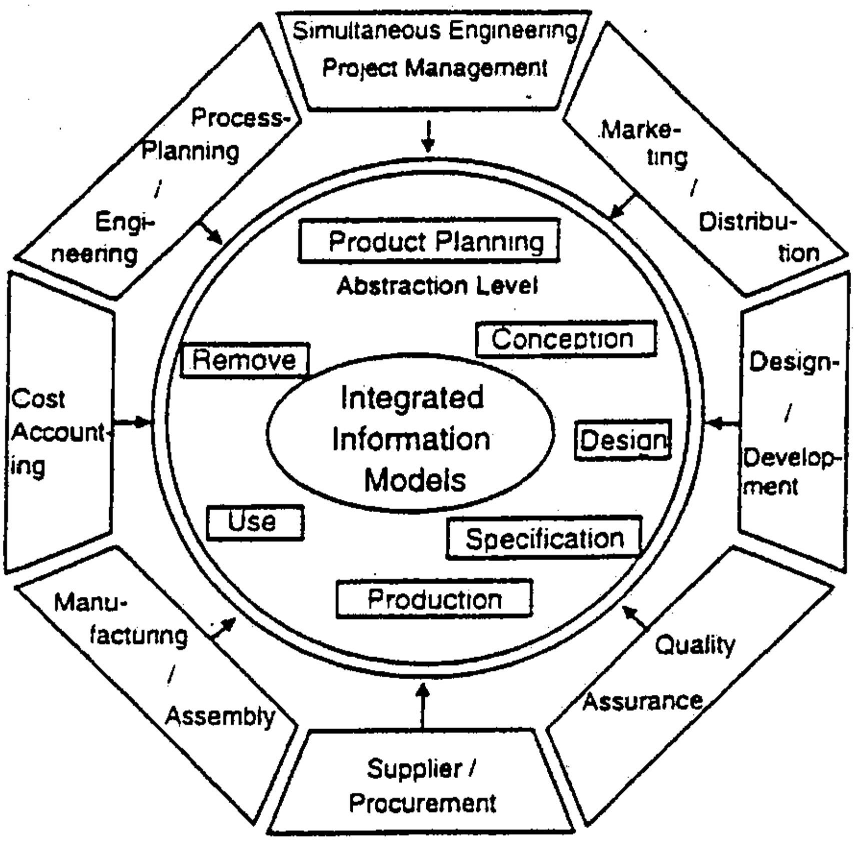



Figure 2.3 Complete Product Life-cycle concerns (Krause *et al.*, 1993)

According to Krause (Krause *et al.*, 1993), a product modelling technology must fulfil the following basic requirements:

- Present actual data: the same information available in different phases of the process and allowance of concurrent access to the data.
- Facilitate product documentation: keep intermediate and final results available.
- Offering decision alternatives: Offer and preserve decision alternatives at early stages to account for those future uncertainties. Support the representation and exploration of process and product alternatives in order to reduce costly iterations and increase product flexibility.

According to Van Der Net, et al., (Van Der Net, *et al.*, 1996) a Product Model must satisfy three basic needs:

- Create a consistent product description for all stages in design and manufacture.
- Capture and record the "design intent".
- Enable manufacturability analysis when designing.

As mentioned above, Krause (Krause *et al.*, 1993) and Van Der Net, et al., (Van Der Net *et al.*, 1996) mention the need for capturing information about product and manufacturing. Section 2.3.2.1 explains the manufacturing information in the Product Model and section 2.3.3 the Manufacturing Model.

(Krause *et al.*, 1993) present a classification of the work done in Product Models, and classify it in the following five categories:

1. *Structure-oriented product models*

The description of the product structure as a product breakdown is the core of structure-oriented product models. This structure can be represented in the form of a bill-of-materials. With this approach, an integrated application of functional modules from different systems is not possible and redundancy of data is unavoidable. Although limited, structure-oriented product models provide some feasible solutions in the product models integration.

2. *Geometry-oriented product models*

Geometric representation of product models using wire frames, surfaces, solids and hybrid models can be used as computer internal models for representing the shape of a product. This representation is used typically as part of basic CAD systems and provides the basis for Numerical Control or Finite Element Method applications. Since these models are created to represent geometry, their extension to non-geometric data is limited.

### 3. *Feature-oriented product models*

The definition of feature provided by De Fazio (De Fazio and The Design Automation Committee, 1993) will be used in this work. *"A feature is any geometric or non-geometric attribute of a part whose presence or dimensions are relevant to the product's or part's function, manufacture, engineering, analysis, use, etc., or whose availability as a primitive or operation facilitates the design process"*

In product models based on the part's geometry the designer can only store some general information, such as the geometry and topology of the product, the position and orientation, and simple relations between these parts. Relevant information is lost in that process and similar information must be created again using analysis programs for grasp planning, motion planning, stability analysis, etc. (Van Holland and Bronsvoort, 1996).

Design and manufacture features can be distinguished. Design features help the designer to communicate ideas. Manufacturing features are defined as the interpretation and combination of form features from the viewpoint of manufacturing, assembly and inspection. Like design features, they are constructed by combining shapes depending on their position in the workpiece.

Bradley and Maropoulos (Bradley and Maropoulos, 1997) identified three main types of features: functional, manufacturing and aesthetic features. The functional features are used mainly in the conceptual design stage, whereas the manufacturing and aesthetic are added in the detailed design stage. At the conceptual and embodiment stages of the design process, feature relations are not completely defined. However, in the redesign activities of the detailed stage, the feature relation allows modifications of the design without changing the structure of the model.

Feature modelling allows the capture of high level information, such as forms, functions, material properties, technological parameters, designer intent, manufacturing precision, etc. This information is essential in the concurrent product and process design (Zha, *et al.*, 1998).

4. *Knowledge-based product models*

These models use AI techniques like object-oriented programming, rule-based reasoning, and constraints systems. Through the employment of these techniques it becomes possible to store human expertise explicitly as well as experience concerning products, processes or factory environments.

The approach followed by Kjellberg and Schmekel (Kjellberg and Schmekel, 1992) use conceptual graphs for represent the Product Model, dividing it in function, physical and solution models. Each of these models represents different aspects of a product. Each model has its own structure, which differs from the others. Between the models are interrelations that can be classified into equality, similarity, dependency or inconsistency. These relations can be implicitly and explicitly defined. The implicitly relations are defined by rules or procedures.

5. *Integrated product models*

These models cover the abilities of geometry, feature, structure and knowledge-oriented models. All types of product information can be stored in the integrated Product Model. Because of the semantic orientation of these systems, the generic product knowledge takes the different stages of the product life cycle into consideration, providing an integrated support of product development during the whole life cycle.

The integrated data models must accomplish the following requirements: Completeness and correctness, minimality and understandability. Other requirements are conceptuality (conceptual models ease the introduction of new implementation technologies), extensibility (allows support for new data requirements without corrupting existing data) and structural integrity (acts as a stable, controlled foundation for design the data model and add extensions) (McKay, *et al.*, 1996).

Bradley and Maropoulos (Bradley and Maropoulos, 1997) proposed the creation of an "aggregate Product Model " to enable integrated aggregate process planning during conceptual and embodiment design stages. In an aggregate Product Model the information modelled should be a subset of the total possible information for the product (Yao, *et al.*, 1998). The requirements for the aggregate Product Model are:

- Support the transition of design data from the uncertain conceptual stage through to the detailed design stage
- Must represent the information in a format suitable for integration with aggregate process planning and design analysis systems.

In order to support the uncertain conceptual stage, the CSG-based model proposed had to be flexible enough to be changed easily and being able to represent in a coherent way the conceptual and the design stages of the product design using the same object constructs. According to the authors, the most suitable is a feature-based representation.

In the Automation and Robotics Laboratory of the University of Massachusetts a system named ProMod was developed (Nnaji, *et al.*, 1993), (Nnaji, 1997). The objective of this system is to develop a system for integrate product modelling and process planning for mechanical components and assemblies. The system is based on features and is capable of performing feature recognition, extraction and interpretation. In the assembly design advisor, the system will be able to check interference and tolerances. For performing this checking, the system considers interference, contact in the spatial relationship, and alignment.

#### 2.3.2.1. Manufacture Information in the Product Model

The decisions about manufacturing made in the design process determine the production cost of the product. These decisions may be more accurately balanced against functional decisions if the appropriate information is available during the early stages of the design process (Chen and Wallace, 1993).

Chen and Wallace (Chen and Wallace, 1993) divide the information that the designer must know about manufacturing in the following classes:

1. Project: Information about the overall design project. Includes information such as the quantity of products, expected production rate and schedule of the project. This information is essential in the clarification of the task and the conceptual design stages, and important in the embodiment design.
2. Material: Information about material properties and availability. Includes information such as the "machinability" and heat treatment of the materials, and sizes, shapes and costs of raw materials.
3. Process: This information is essential in the embodiment and detail design stages and is divided in:
   a) Information about the manufacturing system: Type of system, material handling methods and available skills
   b) Information about manufacturing methods: Machines, tools and operating information.

Yao, et al., (Yao *et al.*, 1998) proposed the creation of an aggregate Product Model capable to provide support in the conceptual and embodiment stages of product design. This Product Model contains information about manufacturing processes, such as welding (Yao *et al.*, 1998), sheet metal processes (Yao, *et al.*, 1999), and tool selection for machining (Baker and Maropoulos, 2000), to support process planning processes.

The need for having information about the facilities that will be used for manufacturing the product can be solved by including this information inside the Product Model, however, this information is particular to the product that is modelled. A different model can be constructed in order to contain information about the manufacturing facilities available in an enterprise. This model is known as "Manufacturing Model", and is explained in the following section.

### 2.3.3. Manufacturing Model

The term Manufacturing Model is used to define, describe and capture the manufacturing situation of a particular enterprise (Young and Bell, 1995), (Molina, 1995) The main objective of the manufacture model is to provide designers and manufacturing engineers with high quality manufacturing information on which to base their decisions.

The Manufacturing Model has been divided in *resources, processes and strategies* in order to achieve general applicability and provide a consistent source of manufacturing information to both users and applications. Resources are the physical elements within a facility that enable the product realisation. Processes explain the use of the Resources.

The use of Resources and Processes provide a consistent representation of the manufacturing facilities and their capabilities. In addition, Strategies are represented as the way in which the decisions are made on the use and organisation of Resources and Processes (MOSES, 1992), (Molina, *et al.*, 1995).

The basic structure for the Manufacturing Model is presented in Figure 2.4 using the Unified Modelling Language (UML), this language is explained in section 3.3.1.3. The structure is divided in functional levels in order to allow the capture of the manufacturing function and behaviour. These levels are Enterprise, Factory, Shop, Cell, and Station Level. This structure is adapted from the Model-Oriented Simultaneous Engineering Systems (MOSES) project, developed as a collaboration between Loughborough and Leeds Universities (Young and Bell, 1995), (Molina, 1995). The *enterprise* class was included to give the Manufacturing Model the possibility of capturing information of more than one factory, as required in the modelling of global enterprises.

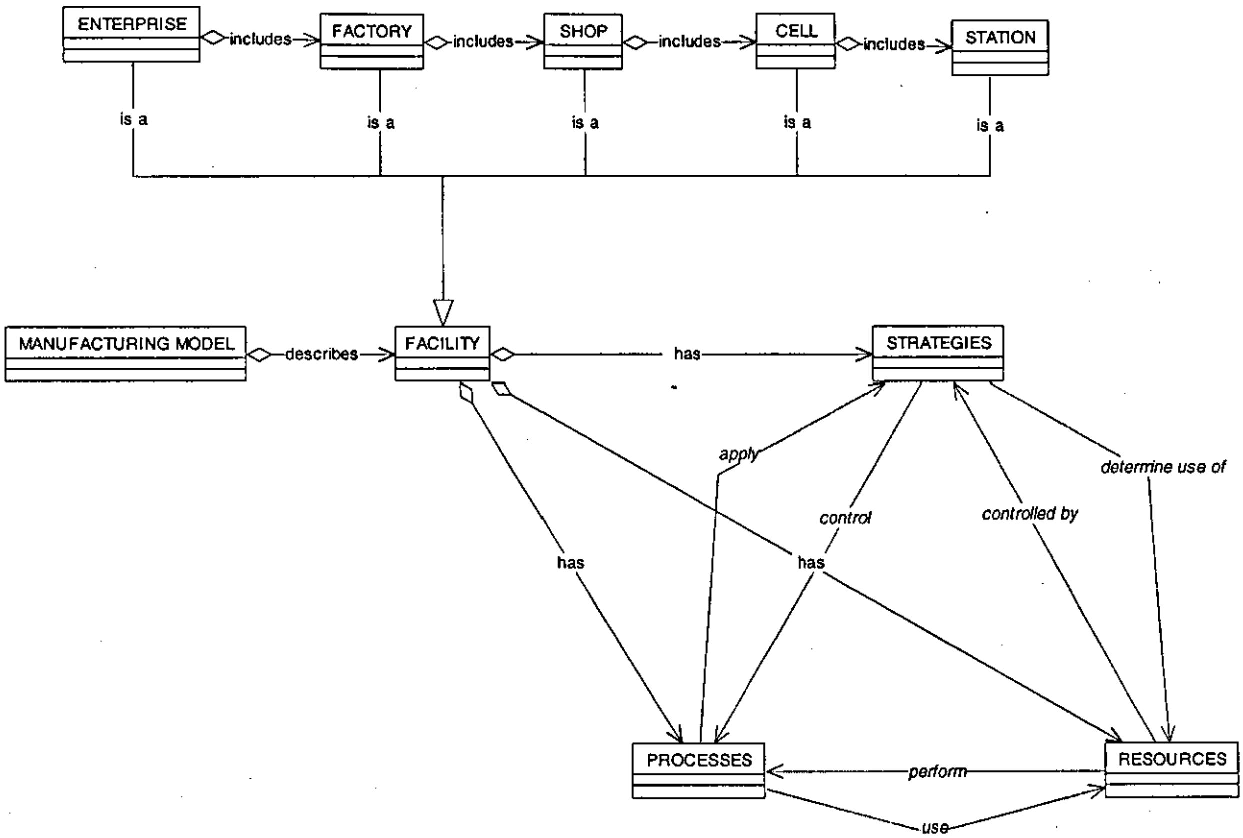



Figure 2.4 Basic structure of the Manufacturing Model using UML notation

The Manufacturing Model in the MOSES project was concentrated mainly in the station level representation issues and in particular the representation of processes and resources and their interactions at this level. That research produced partial implementations of Manufacturing Models related to machining facilities and to the injection moulding process (Young and Bell, 1995), (Molina *et al.*, 1994).

The Manufacturing Model holds information to support the whole product life cycle, allowing the sharing of common data between software applications. The general architecture of the MOSES project is represented in Figure 2.5.

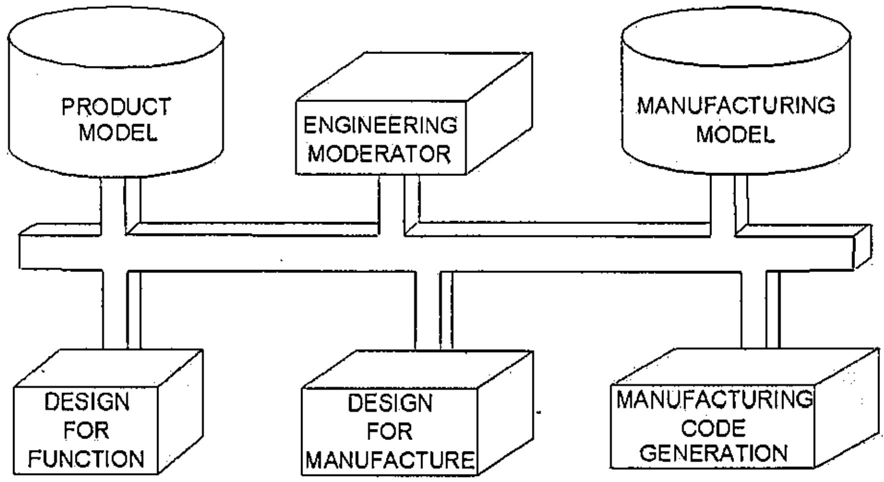



Figure 2.5 MOSES conceptual architecture

Other researchers have identified the need for storing information about the facilities and the processes separate from the product information. These approaches propose to have separate databases for storing information of the processes and the facilities. Some of these approaches are explained below.

Tonshoff and Zwick (Tonshoff and Zwick, 1998) proposed the creation of an Integrated Product and Process Model (IPPM), consisting on a Product Model, a Process Model and an Overall Process Schema. The purpose of their Process Model is to store the information about production processes and machines. The Overall Process Schema is a link between the products and the production processes. This structure is very similar to the one proposed in the MOSES project, however, their process model does not include strategies.

One area of data representation, which has received relatively little attention, when compared to product data, it is process data. The goal of the Process Specification Language (PSL) project developed at NIST is to create a common language for all the manufacturing applications and be able to represent the process information for any given application, not only including manufacturing processes, but also assembly and business processes (Schlenoff, *et al.*, 1997). Figure 2.6 shows data relationships in PSL.

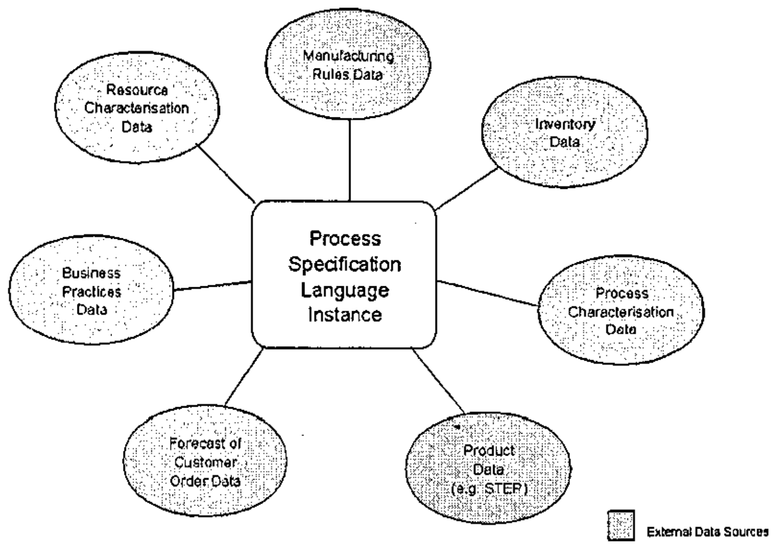



Figure 2.6 Sample Data Relationships in PSL (Schlenoff *et al.*, 1997)

Maropoulos (Maropoulos, 1995) proposes an aggregate process planning architecture that uses a Products Database, Aggregate Process Models and a Factory Model. The aggregate Process Models include information about machining, fabrication, and assembly processes; from which machining is the more developed. The Factory Model includes information about the layout, machines, quality, and suppliers.

The main difference between the approaches proposed by the research group in Loughborough University (MOSES and MIM projects) is that the Manufacturing Model comprises the Resources, Processes and Strategies, while other approaches propose to have separate models, one for the resources information and other for the processes.

### 2.3.4. Standards for Modelling Data

In order to capture the requirements that must be contained in information models several approaches have been proposed based on reference architectures. Some of them have become standards, like CIM-OSA (Computer Integrated Manufacturing – Open System Architecture) (Jorysz and Vernadat, 1990) and RM-ODP (Reference Model for Open Distributed Processing) (Toh, 1999). Other important methods are the Grai Integrated Methodology (Doumeingts, 1998) and the Purdue Enterprise Reference Architecture (PERA) (Williams, 1998).

CIM-OSA focuses on modelling the complete enterprise from requirement and specification through system design, implementation, operation and maintenance. RM-ODP is used for software systems development. They can provide a clear picture of the organisation and assist in the definition of the information needs. They can also be used to evaluate the impact of decisions in the different areas of the enterprise (Prasad, 1996).

ISO STANDARDS FOR MODELLING DATA

In the 1970's the problem of exchanging information between dissimilar CAD systems and between CAD and CAM systems was identified and a number of formats for exchanging information became available, such as IGES, SET, DXF, VDAFS, ISIF, and EDIF. Efforts for standardise the information data exchange has been done by the ISO since 1984. TC184/SC4 is the group that is developing standards for providing the capabilities to describe and manage product data throughout the life of the product (Fowler, 1995), (Bloor and Owen, 1991), (Molina, 1995), (International Standards Organisation, 1997).

The work of the TC184/SC4 group is divided in four areas:

- ISO 10303: Industrial Automation Systems and Integration – Product Data Representation and Exchange (STEP), explained in section 2.3.4.1.
- ISO 13584: Industrial Automation Systems and Integration – Parts Library.
- ISO 14959: Industrial Automation Systems and Integration – Parametric, developed to bring STEP into line with parametric constraint-based CAD systems (International Standards Organisation, 1997).
- ISO 15531: Industrial Automation Systems and Integration – Industrial Manufacturing Management Data. (MANDATE), explained in section 2.3.4.2.

#### 2.3.4.1. STEP

STEP (Standard for the Exchange of Product Model Data), ISO standard 10303, defines a neutral data format for the representation and exchange of product data. The goal of this standard is to complete a system-independent representation of all product-related data during the product life cycle (Krause *et al.*, 1993), (Ashworth, *et al.*, 1996). The technical committee ISO TC184/SC4 is working on the development of a standard structure for a Product Model (NIST, 1997). The methods that are commonly used in the generation of STEP information models are IDEF0 and the Express language. The basic concepts of IDEF0 are given in chapter 3, section 3.3.1.1.

The requirements of STEP are (Vernadat, 1994):

- Support a neutral definition of product and process information.
- Support exchange of product information with the minimum of human interpretation.
- Interrelate a broad range of product information to support applications found throughout the product life cycle.

STEP is organised as a series or parts:

- Introductory parts
- Description Methods
- Integrated Resources
- Application Protocols
- Abstract Set Suites
- Implementation Methods
- Conformance Testing

The relations between STEP and Assembly are explained in section 2.4.5.6.

#### 2.3.4.2. MANDATE

MANDATE is the standard ISO 15531 "Industrial Automation Systems and Integration - MANufacturing management DATa Exchange". In the International Standards Organisation, MANDATE is the working group, parallel to STEP, concerned with "method and development to standardise data that which express information exchanged inside industrial manufacturing plants, except for product definition data" (MANDATE, 1998).

The scope of MANDATE is: *"ISO 15531 specifies the characteristics for a representation of manufacturing management information over the entire industrial process with the necessary mechanisms and definitions to enable manufacturing management data to be shared and exchanged within the factory, with other plants or with other companies"* (MANDATE, 1998).

Inside the scope of this standard are: (International Standards Organisation, 1997)

- *The representation of production and resources information. (Including capacity, monitoring, maintenance constraints and control.*
- *The exchange and sharing of production information and resources information including storing, transferring, accessing and archiving.*

The following are outside of the scope of this standard: (International Standards Organisation, 1997)

- *Enterprise modelling*
- *Product data representation and exchange of product information*
- *Component data (parts library)*
- *Cutting tools*
- *Technical maintenance information.*

ISO 15531 is structured in three series closely related to each other: (International Standards Organisation, 1997)

1. Production data: external exchanges. It is related to the information that must be exchanged with the external environment (customers and suppliers). The aim is to model the production information exchanged between industrial companies in order to improve the exchange and its integration.
2. Manufacturing resources management data. It refers to the resource usage management, such as resource configuration and capabilities, operation management of manufacturing devices, installation and facilities. They also include quality features, maintenance features and safety features.
3. Manufacturing flow management data. This part refers to the material flow control and provides a standard for the representation of data and elements for supporting the flow of material.

## 2.4. Assembly-Related Information in the Product and Process Design Cycle

Reducing costs in assembly in the product design stages can be achieved in two ways (Oh, *et al.*, 1995):

Design for Assembly Analysis: Follows the traditional sequential method in which design for assembly techniques are applied as soon as the product specifications are defined in detail, this approach could be considered as a 'post-design' analysis.

Assembly Representation: This focuses on the development of models of product and assembly planning. The models are usually done by means of solid modelling and graph representation, which are built progressively through the design cycle. Other aspect is the assembly planning, which includes the generation of assembly sequences, and assembly process planning.

Different approaches that have been followed for Design for Assembly are explained in section 2.4.1. As mentioned before, Assembly Representation includes Assembly Sequence Generation, which is outside the scope of this work, and Assembly Process Planning, whose different approaches are explained in section 2.4.3. Section 2.4.5 provides an insight into Assembly Modelling.

### 2.4.1. Design For Assembly Analysis

The main purpose of Design For Assembly (DFA) is to reduce the cost of the assembly by optimising the design of the product. The variables that affect the cost of an assembly are due to the parts themselves and to the assembly operations.

If the product is not designed taking into account considerations for manufacturing and assembly, DFA and other techniques for increasing productivity will be useful only to overcome bad designs. If the product redesign is made during the last stages of design, the changes will not have a great impact on the cost of the product, as explained in section 2.2. According to (O'Grady and Oh, 1991) a saving of 15% to 70% can be achieved by designing a product with assembly firmly in mind.

One of the main rules in DFA is the reduction of the number of parts. When it is reduced, it does not only reduce the assembly costs, but also affects all the operations related with the parts; the cost of producing and maintain all the information related to each part, administrative cost of negotiating with the providers, and planning costs (Kim and Bekey, 1994), (Herbertsson and Johansson, 1993).

When a part is eliminated, its functionality has to be taken by another part, or some parts can be joined in order to have a multifunctional part. However, the cost of producing this parts with increased complexity is normally higher. Suarez (Suarez and Lee, 1997) states that in order to reduce the cost of the product, a balance must be found between the parts and the assembly operations.

#### 2.4.1.1. Design For Assembly Methods

Some authors (O'Grady and Oh, 1991), (Sackett and Holbrook, 1988) provide classifications for the different approaches to design for assembly, here the methods are classified in five groups:

##### 2.4.1.1.1. Specific assembly operation theories

These theories focus on the study of design rules, analysing the part's geometry in order to determine the ease of assembly by converting the part geometry in a value for ease of assembly. This approach has been applied to improve insertion and to avoid tangling. The use of these theories for a complete DFA analysis is not possible, but they can be included as parts of other systems.

##### 2.4.1.1.2. Axiomatic approaches:

These approaches provide general rules and guidelines for design and provide a starting point for designing for assembly. These rules have been derived empirically from years of experience in design and assembly operations. However, they don't provide a quantitative evaluation of the ease of assembly and it is not possible to estimate the improvement due to the elimination or redesign of a part (Kim, *et al.*, 1992).

##### 2.4.1.1.3. Unstructured rules and guidelines

These only provide general rules and guidelines for design for assembly. Several authors had proposed rules (O'Grady and Oh, 1991), (Sackett and Holbrook, 1988), (Owen, 1985), (Rampersad, 1995c), (Bralla, 1986). An example is shown in Figure 2.7.

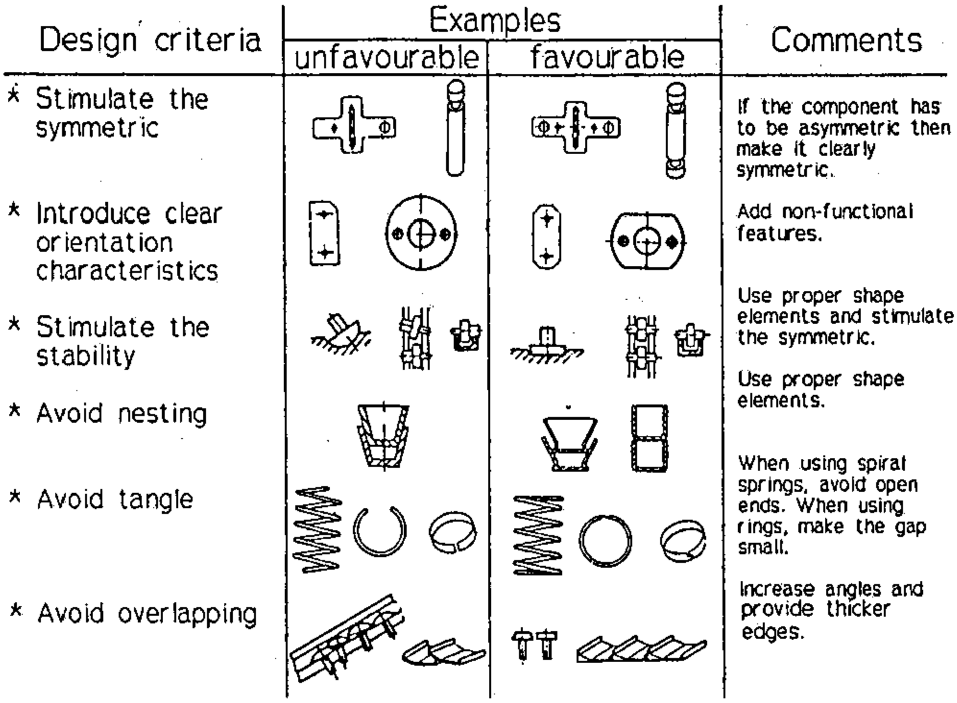



Figure 2.7 Example of unstructured rules and guidelines, from (Rampersad, 1995c)

The application of these rules is useful, but is very easy to find contradictions between them. The application of these rules can lead to more expensive or complex designs, so the DFA must have to be done in a systematic way.

2.4.1.1.4. Structured application of rules and guidelines

These are systematic procedures that assess the assemblability factors when the design of a product is finished. Some of the systems have been converted into software, and are commercially available.

Some of the available systems are:

Hitachi's "AEM" (Assemblability Evaluation Method) (Miyakawa, *et al.*, 1990), (Miyakawa and Ohashi, 1986). This method is based on the evaluation of the insertion process, by comparing the movements required with the basic top-down movement in a straight line.

Boothroyd-Dewhurst DFMA (Design for Manufacturing and Assembly) (Boothroyd and Alting, 1992), (Boothroyd, *et al.*, 1994), (Leaney and Wittenberg, 1992). This method is the most extended, and is available as a software application. The method is based on the reduction of parts and the evaluation of the handling and insertion operations required for assembling a part.

Lucas Industries, Inc. "DFA" (Design for Assembly) (Boothroyd and Alting, 1992), (Leaney and Wittenberg, 1992). This method is based on the evaluation of parts to determine which can be eliminated. By answering a set of questions about the product, the assemblability evaluation is performed The evaluation results are given in an assembly sequence diagram. This method has been incorporated into the TeamSet software, which is a set of tools for concurrent engineering, including QFD, DFA, Manufacturing Analysis, Failure Modes and Effects Analysis (FMEA), Design to Target Cost, and Controlled Concept Convergence.

Poli's "Assembly Analysis and Line Balancing" (Poli, *et al.*, 1986). This is based on a spreadsheet with questions that help to determine how the assembly factors are addressed. These factors include fastening, insertion, handling, and symmetry.

Of these methods, the most widely used is the Boothroyd method. In (Leaney and Wittenberg, 1992) a comparison of the Boothroyd, Lucas and Hitachi methods was made, concluding that the design efficiency of the Lucas method is based on the scope for reducing the number of parts. The efficiency in the Boothroyd method is based on the reduction of the number of parts and in the improvement of the handling and insertion processes. In the Hitachi method, the efficiency is based only in the insertion processes.

2.4.1.1.5. Artificial Intelligence-Based Approaches:

Some research in this area is related to artificial intelligence, The structured rules have been included in expert systems capable of taking decisions based on previous experiences stored on the system.

***Approaches that used rule-based systems*** (Sackett and Holbrook, 1988)

ADAM: Assisted Design for Assembly and Manufacture, in which information about minimising the number of parts, rationalising and guidelines for insertion can be obtained.

DACON: Design for Assembly Consultation, provides an interface for CAD and develops a design evaluation.

PACIES: Part Code Identification Expert System, create three digit codes based on shape, features and part symmetry. This system consists of a set of 163 rules.

***Approaches that use Artificial Intelligence Constraint Nets***

A constraint network is a collection of constraints that are interconnected through shared objects. The value of an object that is linked to a constraint may influence de values of other linked objects, in this way, changes are propagated throughout the network.

Several methodologies and systems have been developed, e.g. IDAERS (integrated design for assembly evaluation and reasoning system), which can provide feedback on the estimated time required for assembling a product (Sturges and Kilani, 1992a), intelligent CAD-DFA (Jakiela and Palambros, 1989), and CAAPP (computer-aided assembly process planning) (Molloy, *et al.*, 1991).

***Approaches that use fuzzy logic***

Zha, et al. (Zha, *et al.*, 1999b), explains the DFAES expert system, based on fuzzy logic to provide the user with ideas and suggestions for improving the design of the product. This system interfaces an AutoCAD drawing with the DFA using a preprocessor tool written in Fortran, which can extract the geometric information of the part to be analysed. The DFA tool uses Boothroyd's approach for analysing the assemblability of the product. In the assembly planning stage, an assembly sequence generator is used to find the most appropriate sequence. In order to store common information between the modules, the system uses a blackboard approach.

#### 2.4.1.2. DFA using Virtual Environments

Virtual Reality can be defined as a synthetic or virtual environment that gives a person a sense of reality. The virtual environments are thought to be the new step in 3D-computer representation. One aspect of design and manufacture that can be significantly affected by virtual reality is design for assembly (Connacher, *et al.*, 1995).

Gupta, et al. (Gupta and Zeltzer, 1995), (Gupta, *et al.*, 1997) proposed a method for estimate the ease of part handling and insertion using multimodal simulation in virtual environments. Their system is called VEDA (Virtual Environment for Design for Assembly). The designer can feel 2D objects and hear sounds when there are collisions among the objects.

There are other virtual environment approaches for assembly, such as the Washington State University's VADE (Virtual Assembly Design Environment). The general idea of this project is to provide the designer with access to manufacturing processes and tools in the form of virtual environments to allow them to "virtually manufacture" the product while designing it (Connacher *et al.*, 1995).

#### 2.4.1.3. Integration of DFA with CAD systems

In order to evaluate the assemblability of the products is necessary to have all the possible information about the parts and the interaction that they will have with other parts. So it is desirable to have a direct link between the CAD system and the DFA tools in order to get directly the information and avoid the need of interpreting and, in some cases, retyping the information. The automatic identification of assembly attributes from a CAD description of a component have been investigated by several researchers, some examples are given below.

An early example of extracting feature information from a CAD system database and then using this information in DFA analysis was presented by Rosario (Rosario and Knight, 1989), in which the geometric information in DXF files are used together with the Boothroyd method for determining the assemblability. Using wire frame representations, the author applies algorithms to determine the symmetry and size of the parts.

Li and Huang (Li and Huang, 1992) have partially developed a framework for an automatic DFA evaluation system. They have developed an assembly features extraction algorithm to convert CAD data into assembly features.

Pham and Dimov (Pham and Dimov, 1999) presented a system for extracting feature-based assembly information from a solid model, capable of downstream assembly planning.

The product DFA/Pro implements the Boothroyd and Dewhurst methodology in the Pro/Engineer CAD system. This system provides a link between the CAD system and the evaluation software, which produces the complete assembly structure, including envelope dimensions for all assembly component parts. This allows the calculation of assembly times, assembly costs, and operation times.

Delchambre (Delchambre, *et al.*, 1996) proposed a CAD method for industrial assembly in which the main stages are Product Structure Analysis, Handling Analysis, Feeding Analysis, Positioning and Insertion Analysis and Joining Analysis. For each of these stages, he proposes some rules and methods for evaluating the design. The Product Structure Analysis can be applied since the configuration and embodiment stages of the design of the product. The other stages are mainly for the embodiment and detailed design stages.

#### 2.4.1.4. Design For Assembly For Large And Heavy Parts.

Over the years, traditional DFA methodologies have proved that can be applied successfully in parts and products that are relatively small in size and weight. The influence of large and heavy parts is considered only in some of the methods, and mainly as a penalty factor that must be added to the factors for small and light parts.

Wong and Sturges (Wong and Sturges, 1994) consider that large and heavy parts are more difficult to assemble due to the weight, inertia for accelerating the mass, moment of inertia caused by rotating the mass and the size of the part relative to its assembly clearances. The motions of large parts can be divided in gross motion and fine motion, the first is related to the acquisition phase and the latter to the assembly.

The results of the tests performed by Wong and Sturges show that parts that are heavier than 4.5 kg introduce problems that reduce the performance in the acquisition phase of the assembly process.

For large assemblies Boothroyd (Boothroyd *et al.*, 1994) states that the acquisition of the individual parts from their storage location will involve significant additional time and because of that the use of his DFA method will "considerably underestimate" total assembly times.

### 2.4.2. Design For Assembly and Concurrent Engineering

The main goal of product design is to obtain a profitable product. The main goal of design for assembly is to reduce the product's number of parts and simplify their assembly. When the number of parts is reduced, according to DFA principles, it is possible that the remaining parts lead to a more expensive product in this case the DFA will not be effective (Chen and Wallace, 1993). That is one of the reasons of the need to perform the analysis in a concurrent engineering environment, taking into account other parameters.

A concurrent engineering platform must allow its users to evaluate designs and explore different alternatives at various levels of abstraction to zoom in on promising design paths (Kim, *et al.*, 1996).

Design for assembly involves the consideration of the assembly process while designing. As such it can be considered as an important aspect of concurrent engineering (Oh *et al.*, 1995). All the assembly related activities, such as design for assembly, assembly sequence generation, assembly process planning and tolerance analysis can be running concurrently in the background, transparent to the designer (Lim *et al.*, 1995). Some researchers recognise the need of the integration between assembly planning and design in a concurrent engineering environment, among them are Kim (Kim *et al.*, 1996), Mo, et al., (Mo *et al.*, 1999), and Delchambre, et al., (Delchambre *et al.*, 1996).

The concurrent design of a product and its assembly processes has enormous potential in improving existing product development practice. As the design of a product and its assembly plan are evolving, the DFA evaluation can provide the necessary feedback to improve the design. The evaluation of an assembly process plan can produce suggestions for the redesign of products and complement DFA techniques. An integrated computer-based environment can help to shorten the product development cycle and to achieve better costs and quality.

### 2.4.3. Assembly Process Planning

An assembly plan describes how to assemble the product. Assembly Process Planning considers the technological requirements of each job to be performed, while Production Planning and Control systems are responsible for planning the utilisation of production resources which are required, over a planning horizon, considering some demand pattern (Cunha, *et al.*, 1999).

Assembly Process Planning is a series of activities related to the engineering data management of assembly operations of the parts that form an assembly. The data is used for the generation of assembly tasks and task sequences of the assembly operations, the estimation of task time, the determination of the resources needed to produce the assembly and the allocation of tasks to workers (Tran and Grewal, 1998). These activities are generally performed manually and are driven by the experience of product planners (Grewal, 1997). This task is becoming increasingly difficult because flexibility is increasing in the manufacturing environment.

For assembly processes planning the following activities are commonly done (Shimizu and Nishiyachi, 1996):

1) Classify the parts into operational groups with reference to the Bill of Materials (BOM) and determining the order of assembly.
2) Calculations of the assembly time for each part with a pre-determined timetable and assess the total assembly time.
3) Create standard work sheets with enough information, diagrams and drawings to perform the assembly activities.

4) Change the pitch time of line and the number of assembly process according to the change of production volume.
5) Create process sheets with enough information for each operator.

Some of the research done in this area is reported below.

**Integrated Part and Assembly Planning – CSIRO, Australia**

Tran and Grewal (Tran and Grewal, 1998), (Tran and Grewal, 1995), proposed the creation of a Data Model for an interactive assembly planning software system based on the interactive generation of sequence of work elements with knowledge-based handling analysis and time estimation and dynamic line balancing. The main stages of the process are task sequence generation, task analysis (includes handling analysis, equipment selection and time estimating) and group tasks (line balancing). The model is shown using UML notation in Figure 2.8.

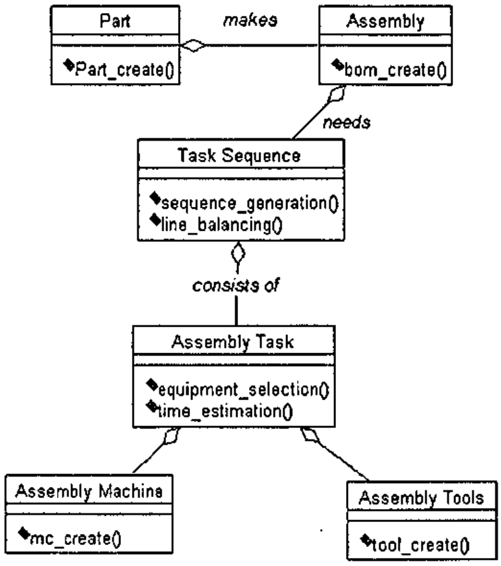



Figure 2.8 UML representation of the model proposed by the CSIRO

**Delft Intelligent Assembly Cell (DIAC) and FLEXFACTORY - Delft University of Technology, Netherlands**

The DIAC project had the aim to produce a flexible assembly cell capable of performing a broad spectrum of assembly tasks. In this research project several areas were explored, one of the research areas within this project is Computer Assembly Process Planning. The decomposition of tasks was considered in four levels of abstraction: batch, product, part and primitive levels.

The input to the sequence planner consists of product data that have been derived from a commercial CAD system. The data is stored in a Product Data Model that consists of the following elements: (Waarts, *et al.*, 1992)

- The geometry of each part, given as a boundary representation, specified relatively to a part-frame
- The Bill of Materials. Its name and its position in the product specify each part. The position of the part-frame is relative to a product-frame.
- Relations between parts in the product. Whenever two parts are connected, they have a relation, which is expressed relatively to a relation-frame.

The FLEXFACTORY project continued the work done in the DIAC project, with the objective of proposing a hardware design of a flexible assembly system in such a manner that it shows a high degree of trustworthiness for industrial application (Vos, 2000).

**Archimedes**

The Archimedes project, under development in the ISRC of the Sandia National Laboratories in the U.S., has the aim of "advance the state of the art in assembly planning technology". The software structure is based on a sequence of modules, each viewing the product at a greater level of detail. The relationships between these modules are represented in Figure 2.9 (Ames, *et al.*, 1995), (Kaufman, *et al.*, 1996).

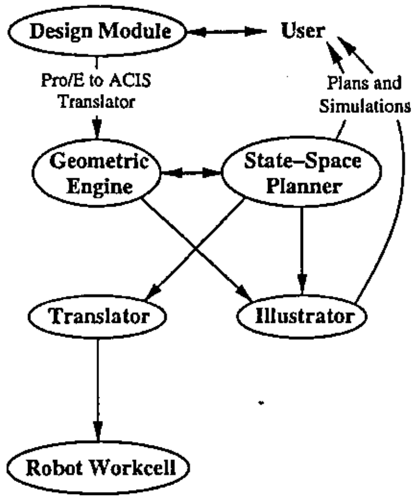



Figure 2.9 Architecture of Archimedes 2 (Kaufman *et al.*, 1996)

The design module of the system was built on Pro/ENGINEER. The geometry engine finds part-part contacts automatically from the CAD data and constructs a non-directional blocking graph of the assembly. The state-space planner performs a search to find the best plan according to a user-specified criterion.

One of the aims of the Sandia's research was to determine an appropriate balance between human and machine planning (Calton, 1999). In the assembly sequence selection, the designer imposes constraints to the automatically generated space of possible assembly sequences, such as defining subassemblies before other parts, declaring preferred directions, etc. However, Ames (Ames *et al.*, 1995) reports that one of the main limitations of the system is the limited interactivity allowed with the user, who can not interact in the choice of assembly plans. Another limitation is the lack of non-geometric data, necessary for industrial-size assemblies.

#### 2.4.3.1. Assembly Sequence Planning

In order to be able to produce an assembly plan, it is necessary to define in the sequence in which the assembly operations will be carried out. Very often, the assembly of a product allows more than one sequence of operations. It is the responsibility of the planner to find the best sequence under a given set of planning constraints, e.g. production volume, technology available, etc.

According to (Lim, *et al.*, 1995), assembly sequence generation for assembly planning has almost reached maturity stage. Several techniques have been developed to aid to generate Assembly Process Planning information using CAD descriptions of a product assembly. An example can be seen in Figure 2.10.

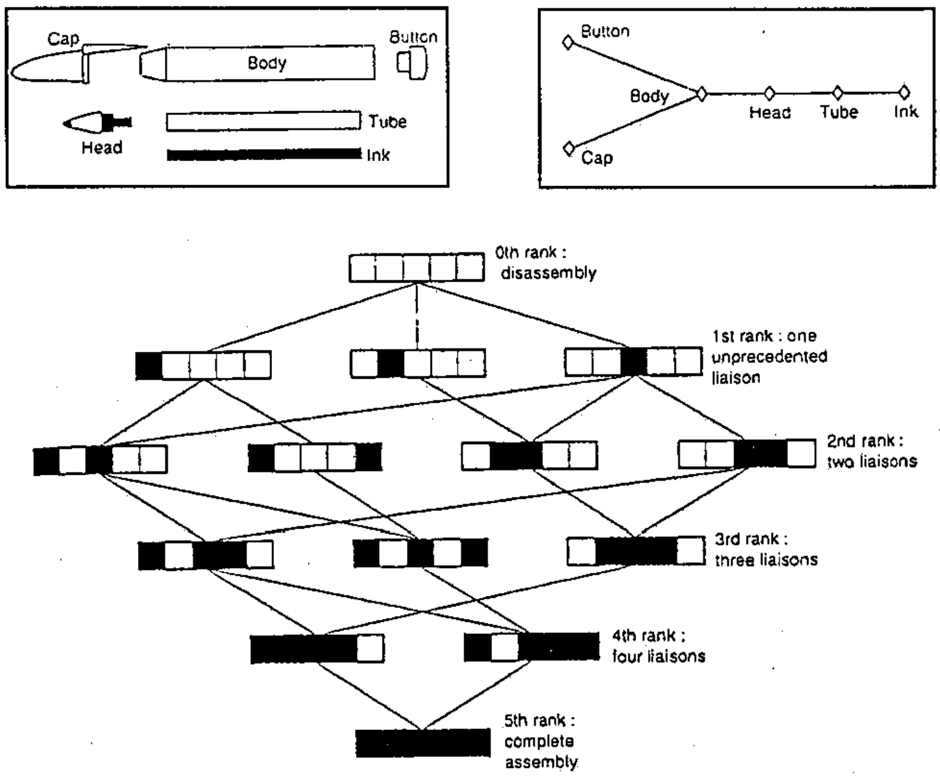



Figure 2.10 Example of assembly sequence generation (Sturges and Kilani, 1992b)

Assembly sequence planning has been studied by several researchers, a review is given by Gottschlich, et al. (Gottschlich, *et al.*, 1994). Lim, et al. (Lim *et al.*, 1995) present a list of research done for determining the assembly sequence through disassembly approaches. Other researchers have employed knowledge-based techniques to encode into rules assembly process planning.

Assembly sequence generation was left out of the scope of this thesis, however, the structures for the information models, proposed by the author in chapter 7, could be extended to support this activity.

### 2.4.4. Relations Between Design For Assembly and Assembly Process Planning

DFA approaches provide guidelines on how to design a part based on previous studies, or evaluates the assemblability of the products. The functions considered are those that include design for flexibility, functional rationalisation, component feeding, gripping and insertion processes, and their structural relationships. However, DFA generally gives no consideration to the assembly plan by which the product will be assembled.

During the planning process, there is a great deal of information that could be useful for the designers, and there is no way of passing that information to them. This information includes selection of assembly techniques and production equipment, assembly production rates, and line balancing. As a result, that information is lost, but if there were a link between the planning process and the design process, and if there was integration, a lot of benefits could be obtained (Hsu, 1993), (Hird, *et al.*, 1988).

Molloy, et al. (Molloy *et al.*, 1991) recognised the importance of performing concurrently Design for Assembly analysis and Assembly Process Planning. In the system proposed by Molloy, an If-then evaluation system is used for applying DFA guidelines to the components and assembly operations. The function of the assembly planning system was to read the information from a CAD system in order to obtain the disassembly sequence.

Hsu, et al. Proposed a method called “FANTASY” (Feedback Assembly iNTegrated Autoredesign System). The system aims to achieve assembly-oriented design trough the feedback evaluation of a given assembly plan (Hsu, 1993). FANTASY is part of the intelligent environment developed at Purdue University. The input to the system is a feasible plan for assembly based on precedence knowledge, this is generated with an Assembly Planning software, also developed in Purdue.

There are certain similarities between the information that a model must contain for performing Design For Assembly and Assembly Planning, but the level of detail is different. For example, there must be the components involved, the relationships between them and the information about the facilities that are available for assembling the product. However, the results from DFA are completely different from those of the Assembly Planning. DFA will tell the designer in which details of the product there is a possibility of redesign and the assemblability efficiency. While in the assembly planning the result will be a detailed assembly plan that can be used directly for production.

### 2.4.5. Assembly Modelling

Assembly modelling is divided in the CAD representation of assembly operations and in models that hold the information about assembly.

#### 2.4.5.1. Solid Modelling of Assemblies in CAD

There is an assumption that tells "If the designer can see something vividly enough, then he/she will know what to do and how to avoid errors". Nowadays most of the main CAD software systems have this 'vivid representation' that allow the designer to identify possible interference problems.

The CAD area is growing very fast, and some capabilities that were available only in large and expensive systems few years ago are now available for PC (Pye, 1997).

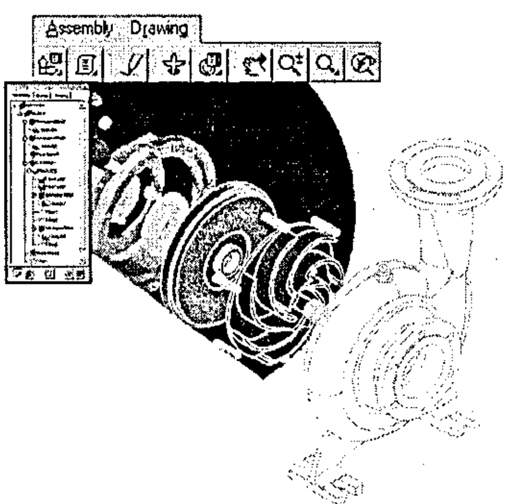



Figure 2.11 Assembly modelling in Mechanical Desktop©

There are two levels of assembly modelling, the first is *design assembly modelling* in which the designer works with a fully accurate solid model, modelling the behaviour of the design with mechanism modelling and finite element analysis software. The other level is modelling the assembly process, which requires interactive movement of parts and sub-assemblies.

Most CAD systems model assemblies .using geometric relationships quantitatively with geometric constraints such as surface-to-surface, point-to-point and axis-to-axis constraints (Kim and Szykman, 1996), (Kim and Szykman, 1998).

The graphical information handled by current CAD systems support applications such as editing of geometry, dynamic interference checking, NC code generation and verification, generation of meshes for Finite Element Analysis. Besides geometric information, some CAD systems have added parametric and feature capabilities (Baxter, 1994). In the assembly aspects, some software are capable of maintain mating conditions such as *mate, orient, align, adjust* between parts. However, those relations are not enough for capture completely the design intent, because some essential aspects of the design are not captured, some examples are given:

- The purpose of the existence of the parts, which is needed for performing DFA analysis.
- In Finite Element Analysis the geometry is used for generating the meshes, but that is far from being enough to perform an engineering analysis, the purpose and working conditions are necessary.
- In Value Analysis one of the aims is to remove non-useful features. Without the knowledge of the reason for the features, the analysis can't be performed
- In QFD a design must be compared against the customer requirements, so the functionality must be captured.

#### 2.4.5.2. Assembly Models

Relatively little attention has been given to design or modelling methodologies for assemblies (Kim and Szykman, 1996), (Kim and Szykman, 1998). Problems arise when modelling tolerance relations and assembly operations; tolerances should be more than just attributes, while assembly operations should surpass the "bill of material" stage (Van Der Net *et al.*, 1996).

An assembly model must be able of capturing the information needed to describe the entities and activities associated with assemblies and assembling. This information must be useful for designers of products, assembly systems and logistic systems, as well as for supplier relations, field support, disassembly and recycling (Whitney, 1996).

A survey of existing literature made by Hsu, et al. (Hsu, 1993) shows that there are three popular representation schemes for modelling an assembly:

- Graphs: Are easy to understand, but there are difficulties in capturing multiple relationships between a pair of nodes.
- Relational Databases: Are based on good theoretical foundations, but is not adequate to represent semantic meanings.
- Object-oriented Databases: Are versatile, provide encapsulation and include inheritance capability. The disadvantages are that are complex, and that they need special software support.

According to Henson (Henson, 1995b), Assembly Models can be classified in

Part position models and solid modelling

Hierarchical models and assembly sequences to aid automatic assembly

Relational models and assembly languages

Relational models and CAE information systems

#### 2.4.5.2.1. Part Position Models and Solid Modelling

Part position models represent assemblies by positioning solid models of individual components in space. These solid model representations are generated by computers from definitions made in a Constructive Solid Geometry (additive and subtractive compositions of primitive building blocks) or by Euler-operator constructions used to create B-reps (boundary representations: collections of faces described by surface equations and bounding edges) (Requicha and Voelcker, 1983).

Besides the geometry of the part, for this model it is necessary to define a 4x4 homogeneous transformation matrix for each part to position it in space. These matrixes position the part's local axes relative to the system's global co-ordinate system. Defining solid models and transformation matrices for each component at the same time in the same space specifies assemblies. Current CAD systems hide the transformation matrixes from the user, and allows the definition of rotations and translations to position parts relative to each other (Baxter, 1994).

Part position models unambiguously represent the geometry of components and their positions in space. To some extent, they can be used for mass and volume calculations and for static interference checking.

#### 2.4.5.2.2. Hierarchical Models and Assembly Sequences

De Fazio and Whitney, (De Fazio and Whitney, 1987), Baldwin, et al. (Baldwin, *et al.*, 1991) and Homem de Mello and Sanderson (Homem de Mello and Sanderson, 1990) developed the use of hierarchical graphs to generate valid assembly sequences. De Fazio and Whitney use a directed graph, while Homem de Mello and Sanderson use AND/OR graphs.

Hierarchical models of assembly are useful for determining assembly sequences and such sequences may be essential for some types of engineering analysis. However, the lack of relational information in the hierarchical model would cause some limitations if used in other applications:

- The user has to supply the information about valid assembly sequences.
- The assembly must be designed and the geometry of individual components established before the hierarchical model can be developed.
- The underlying theory assumes that assembly is the reverse of disassembly.
- They do not capture the design intent of the engineer.

Baldwin et al. (Baldwin *et al.*, 1991) followed this approach to find and represent all geometrical and mechanical assembly constraints as precedence relations. This method analyses the liaison diagram and generates all the possible assembly sequences.

Eng, et al. (Eng, *et al.*, 1999) proposed a system for assembly sequence generation based on features using a disassembly approach. By knowing the degrees of freedom that exist between the mating surfaces of two parts, they generate a matrix that gives a high level definition of the features. The free face heuristic test, used together with a 'disassemblability' test, and the bounding box conditions helps in the definition of the assembly sequences.

For complex assemblies De Fazio, et al. (De Fazio, *et al.*, 1997) are using Genetic Algorithms for explore many subassembly partitions and sequence choices.

2.4.5.2.3. Assembly Languages for Programming Robots

In the 1970's research was made on how best to program assembly robots. At the time, they could either teach the robots the tasks or code the robots with the transformation matrices that describe the robots' paths. Systems such as AUTOPASS (Lieberman and Wesley, 1977) were developed to simplify the programming of assembly robots. These languages use English-like statements such as PLACE, TURN, and INSERT.

2.4.5.2.4. Relational Models and CAE Information Systems

In a relational model, each node represents a component of the assembly and the arcs represent the relationships between the components. The relationship has to constrain the mating components sufficiently to allow the computer to manipulate the model in meaningful ways.

Lee and Gossard (Lee and Gossard, 1985) present a relational model, which is created interactively, and introduce the concept of the virtual link. A virtual link is the complete set of information required to describe the relationship and the mating features between a mating pair. The relationships provided by the virtual link are *against, fits, rigid attachment, conditional attachment, translational constraint and rotational constraint.*

The difference between Top-down and bottom up assembly modelling is that the first is based on generating a functional or symbolic description of a prospective design, (conceptual synthesis of a new system) and performing a refinement of component geometry. The functional model should be validated to before moving into individual part design. The bottom-up approach is used when the detailed designs of the parts of the assembly are available. It starts with component design (with a mental model of the design) followed by continuous revision of the mental models and part design. Designers, in applying the alternate use of abstraction and refinement, mix both top-down and bottom-up approaches in assembly design (Kim and Szykman, 1996), (Kim and Szykman, 1998), (Shah and Tadepalli, 1992).

***Mantyla's top-down modelling system (Mantyla, 1990)***

Mantyla developed a system that aims to aid design engineers early in the design process. Based in the top-down approach the goal is to develop modelling techniques that can support design of assemblies.

Mantyla's aims are summarised by his requirements for a modelling system:

- Provide support for representing the assembly information at different levels of abstraction.
- Represent design intent by preserving the reason for an entity existence.
- Make explicit the level of commitment of the designer to properties of the design. In particular, strongly committed details must be treated as constraints for subsequent design phases.
- Support redesign in various ways, for example, by supporting parallel refinements of a single started model
- Provide documentation tools for capturing functional specifications for use by later design phases and for redesign.

**NIST Repository for Design, Process Planning and Assembly**

Regli and Gaines (Regli, 1997) present a repository developed with the sponsorship of the National Institute of Standards and Technology (NIST) in the United States. The main goal of this repository is to give researchers access to problems taken from industry. It also provides a point for collaboration, allowing researchers to post challenge problems to a wide audience, share results or perform larger-scale experiments.

This repository is intended to provide information for design, process planning and assembly. The repository contains parts in several data exchange formats, and covers a variety of commonly used manufacturing processes. In this repository Assembly Planning has been approached as both a geometric and symbolic reasoning problem. Geometric reasoning is applied to motion planning, fixturing, and robotic and human grasping. Symbolic reasoning emphasises the generation of assembly sequences for individual components and subassemblies.

At present in the assembly area, the repository only contains some examples of vendor demonstrations and a few products. The web address for accessing the repository is: http://repos.mcs.drexel.edu/

#### 2.4.5.3. Semantic Nets, Entity-Relationship Modelling

The entity-relationship modelling has the following components:

- Entities: Set of objects with common properties
- Attributes: Characteristic of an entity
- Relationships: Association between two entities such as 'part-of' or 'enclosed-by'

In semantic models, the models can act as means of communication between different engineering disciplines. The analysis of semantic models of different assemblies allows to design effective data structures for design representation. The simple data structures that are derived from geometrical representations are increasingly inefficient for the type of design processing required.

Taleb-Bendiab (Taleb-Bendiab, *et al.*, 1993) developed a system that models axisymmetric assemblies using the entity-relationship semantic data modelling method.

#### 2.4.5.4. Functional Models of Assemblies

During the design process the designer gives constraints to the geometry in order to realise functions and subfunctions. There are two main ways for representing the functions in the domain of mechanical products: syntactic languages and models of the flows between the inputs and outputs of a product.

Descriptive syntactic languages use grammar to describe a product, in which nouns are related by verbs. Some of the verbs used for representing assembly operations are: contain, convey, convert, control, house, hold, cover, constraint, clamp, couple, pivot, support, locate, drive, guide, limit, seal, fasten, lubricate, strengthen (Baxter, 1994).

Flow type representations model assemblies in terms of the inputs and outputs to and from functions, they describe how a device is intended to work (Baxter, 1994).

2.4.5.5. Assembly Features

Feature models are widely accepted in design and manufacture, and several trends in assembly modelling are using them to represent the information, (Shah and Rogers, 1993), (Shah and Tadepalli, 1992), (Van Holland and Bronsvoort, 1996), (Delchambre, *et al.*, 1989), (Whitney, 1996), (Deneux, 1997), (Molloy, *et al.*, 1993), (Sodhi and Turner, 1993), and (You and Chiu, 1996).

Mascle (Mascle, 1999) gives the following definition: *"An assembly feature is defined using any geometric (a sub-face) or technological (a process) or functional (describes the product's function as an allowance or a liaison) information assigned to a face, a part or a subassembly".*

Van Holland W., Bronsvoort W.F. (Van Holland and Bronsvoort, 1996) distinguish two types of assembly features: handling and connection features.

> ***Handling features*** contain information on how to handle a component and grippers and fixtures can be specified.
>
> ***Connection features*** contain information on a connection between components, including tolerances. Some of these features only exist for establishing a connection with a feature on another part, e.g. a pin in one part and a hole on the other. Usually both parts are modelled independently, and by doing so the relation between both parts doesn't exist. In order to establish directly the relation between the parts is necessary to model the parts simultaneously.

Some research that have been done in the assembly features area are:

DeFazio et al. (De Fazio and The Design Automation Committee, 1993) proposed a model called "Feature-based Design for Assembly", in which the features proposed are: *Part features* (location, volume, bounding box, material, instances), part's shape features (location, orientation, dimensions, tolerances, surface texture, threaded surfaces, etc.) and *assembled product's features* (liaisons between parts and through which features, degrees of freedom, distance between mating features). The assembly sequence generator uses the feature mates information and using a question-answer dialog with the user, the system stores the feasible assembly sequences. The sequences can be edited for eliminating assembly states, actions and plans (De Fazio and The Design Automation Committee, 1993).

Delchambre uses a polyhedral representation of objects and parts. In his model he represents the following assembly features: *parts' geometric features* (shapes, dimensions, relative position within the final assembly), *physical features* (handibility, base path and begin-end priorities), and *topological features of the assembled product* (recognition of similar parts, types of contact, interference and assembly constraints).

Shah's work (Shah and Tadepalli, 1992) is a bottom-up approach that deals with the determination of geometric constraints: degrees of freedom, compatibility between mating features, orientation and insertion limits. The information that defines an assembly feature must have parameters and rules. The parameters can be dependent or independent. The dependent parameters are those that are constrained by something already existent in the model. The independent parameters are user choices at the time of the assembly. The derivation of dependent parameters has to be specified by procedures of inheritance rules.

Mascle, et al. (Mascle, 1999) proposed a system called SCAP (Système de Caractéristiques d'Assemblage de Produits – Product Assembly Features System) to support assembly. The activities supported by SCAP are components modelling, assembly sequences generation, assembly resources, evaluation and assembly planning.

#### 2.4.5.6. STEP and Assembly

STEP considers assemblies as lists of parts without reference to physical or functional connectivity and can be used to generate bills of materials (Henson, 1995b).

Liu and Fischer (Liu and Fischer, 1993) used STEP, EXPRESS and ROSE for building an assembly schema. They illustrate the schema with an example based on a toy.

In STEP there is an entity called assembly_model, but the assembly relations remains undefined. The assembly_model in STEP is designed to carry an assembly system configuration in a hierarchical structure.

The integrated generic resources and application resources defined in the Parts of the STEP can describe most of the information mentioned above. Part 44 Product Structure Configuration Model provides a mechanism to represent hierarchical structures among assemblies, subassemblies and parts, however, tolerances for relative position, connection relations between components and assembly features of the components have not yet been considered in the STEP models.

Yin, et al. (Yin, *et al.*, 1999) proposed the creation of a system based on STEP for performing Assembly Process Planning. The central part of the system is the assembly model in which the geometric information of the product is stored together with information about the relations between the parts. The Product Model consists of product objects, subassembly objects, part objects and connector objects. This information is used to generate the feasible assembly sequences, which are evaluated for optimisation.

### 2.4.5.7. Integral Assembly Model

Integral Assembly Models consider the effects that changes in one area of the model have on other areas. If a change is done in the product's design, it is very likely that a change will have to be done in the system or in the processes, and also, if the change is done in the system, the product design is likely to need some changes.

Andreasen (Andreasen and Ahm, 1986) proposed to conduct DFA for the Product in the assortment, structure and parts levels, and for the Assembly he proposes the system and the operation levels. There are strong relations between the levels in the product and in the assembly. The product structure has great influence on the layout of the system, etc. The conclusion is that DFA must be conducted at all levels of the product. The decisions taken as a result of a DFA analysis not only affect the product, but also other areas in the company (Herbertsson and Johansson, 1993).

For obtaining an integral assembly process, Rampersad (Rampersad, 1995a), (Rampersad, 1995c), (Rampersad, 1995b), (Rampersad, 1996), proposed a division of the variables in the assembly model into product, assembly process and assembly system. Each of these variables consists of three levels. The arrows in Figure 2.12 show the interaction between the different variables and levels.

In the case of a change in one or more of the variables, the other variables have to change. The properties of the assembly process can be regarded as a function of the properties of the product and the assembly system:

*<assembly process> = f(<product>,<assembly system>)*

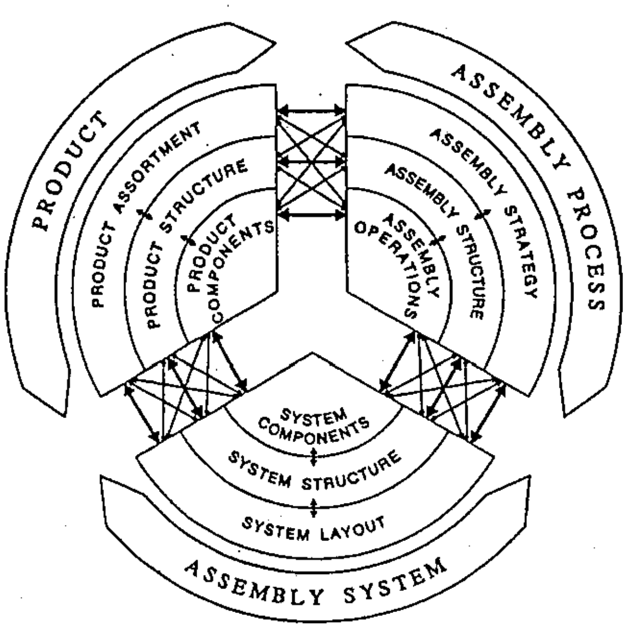



Figure 2.12 Interactions in the integral assembly model (Rampersad, 1995a)

The models proposed by Sturges and Kilani include the following levels, illustrated in Figure 2.13 (Sturges and Kilani, 1992a), (Lim *et al.*, 1995).

- Component-level model. Contains information about the parts in a product assembly.
- System-level model. Information about the interactions of the parts and assembly sequences for DFA evaluation
- Assembly-process-level model. Relates to the physical facilities needed for an assembly process.

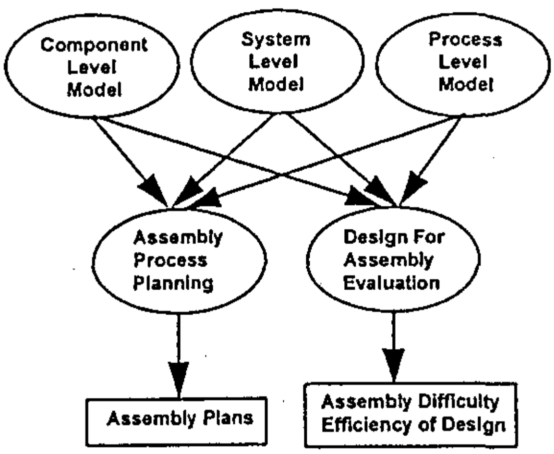



Figure 2.13 Component, System and Process Level models (Lim *et al.*, 1995).

Another integrated approach is the Integrated Part and Assembly Planning work developed in the CSIRO, which was explained in section 2.4.3. This approach is illustrated in Figure 2.14.

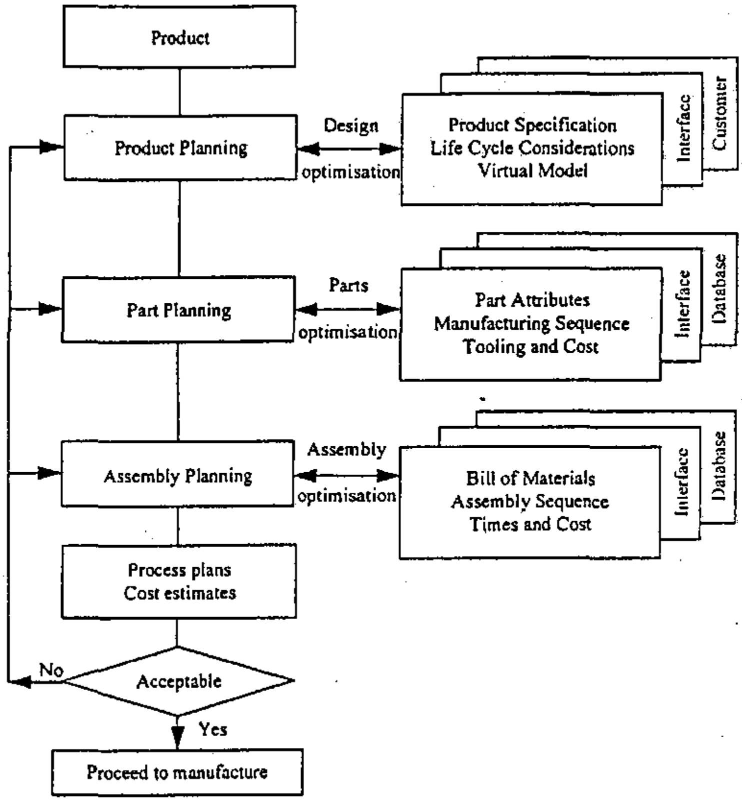



Figure 2.14 Integrated product and process planning, CSIRO approach (Grewal, 1997)

In the IDAERS (Integrated Design for Assembly Evaluation and Reasoning System) software proposed by Sturges and Kilani (Sturges and Kilani, 1992b), the factors that affect the assemblability are identified. These factors are differentiated into component-level factors, system-level factors and process level factors. The first are obtained with a feature-extraction operation from the individual components. The system level factors depend on the interactions between components and can be identified with an assembly sequence generator. The process-level factors depend on the process used to assemble the product. Once the factors are identified, the difficulty associated with each model is assessed.

In the long-term Integrated Design and Assembly Planning (IDAP) project, (Richter, *et al.*, 1989), (Hird *et al.*, 1988) the objective was to create an integrated system for design and assembly process planning by using models to represent the product, the assembly system and the production operation, within the framework of the IDAP-System.

According to Seidel and Swift (Seidel and Swift, 1989), the IDAP system should have two basic views, one directed to the designer and other to the assembly planner. The area directed to the designer covers aspects such as functional relevance manufacturing costs, feeding and gripping. The designer can describe the assembly structure in a graphical way, covering assembly (insertion) processes, non-assembly processes and work holding processes. The assembly engineer can use the same view, or change to a precedence graph representation of the assembly structure. The engineer can add operations, rearrange relations between operations, assign assembly equipment, standard times, etc.

In the attempt to integrate IDAP within a manufacturing enterprise, the project has highlighted a critical need to restructure an organisation in order to support the design and planning activities in a co-operative manner (Lim *et al.*, 1995).

### 2.4.6. Assembly in the Product and Manufacturing Models

#### 2.4.6.1. Assembly Approach in the MOSES Project.

The objective of the assembly data work package of the MOSES project was to develop a data model that could represent mechanical assemblies in a distributed working environment where data is shared between applications. This support had to be done throughout the product's life cycle (MOSES, 1994b), (Henson, 1995a), (Henson, 1993).

In the MOSES project, assembly models were defined as "a constituent of information systems that will provide the product and manufacturing data necessary to integrate product design and process design" (Henson, 1995a),
In projects previous to MOSES made at Leeds University, the product data framework developed in the "Structure Editor" (Henson, 1993) considered assemblies as lists of parts without reference to physical or functional connectivity. This was suitable for bill of material (BOM) applications but is of little use to applications where geometry is important. One of the goals for the MOSES project was to create new structures to support information about assemblies required by applications throughout the life cycle of the product.

The proposal for the assembly related area of the MOSES project followed the work of Mantyla (Mantyla, 1990), mentioned in section 2.4.5.2.4, about the need for providing information to support the assembly information at different levels of abstraction. It is important to note that the ability to represent information about assemblies at different levels of abstraction implies the ability to represent incomplete and inconsistent information about the functions and approximate geometry of components.

Henson (Henson, 1995a) stated that the MOSES project would not try to follow the work done in DFA because it would not have contributed to the MOSES concept of information sharing. The assembly data model developed by the MOSES project had to be able to support enquiries about components' degrees of freedom, the reason for their material, and their assemblability.

The research in the assembly modelling area of MOSES followed two paths: Support for assembly mating conditions neutral of specific assembly languages, and development of a data model to support the description of behaviour.

The results of this area of research in the MOSES project were reported by Henson (Henson, 1995a), indicating that a data model to support the description of mating conditions was developed relying on the description of the degrees of freedom. That data model was not implemented and only a statement of information requirements was produced. In the description of behaviour, the bond graph notation was selected for describing it. A database and applications were developed to demonstrate that descriptions of structure and behaviour were, to some extent, integrated. Furthermore, a DFA application was not implemented because the researchers believed that showing the assembly data model in the context of analysis would be more meaningful to the collaborators of the project.

In the report for the assembly area of the MOSES project (Henson, 1995a) the following further research is proposed:

- Solve issues about the integration of different product structures (such as function, physical structure and behaviour).
- Management of constraints between different product structures.
- Is necessary to create tools for populating the different product structures of the Product Model in a way that reflect existing design processes.
- The data model for assembly mating conditions should be applied and exploited.

The Manufacturing Model area of the MOSES project, was focused on machining and plastic injection processes. Processes, resources, and strategies related to assembly were not included in that research.

2.4.6.2. Functional Data Model for Assemblies

Elmaraghy, et al., (ElMaraghy, *et al.*, 1993) presented a prototype system for function-oriented feature-based modelling. This system is based on the assumption that higher level functional features (keyseat, fillet, chamfer, etc.) are more natural for designers than form features. In addition to geometric and topological information, functional features could contain information about the function they have, in relation with other features, tolerances, materials, etc. In the prototype system, the product is represented by a hierarchical data structure from the highest assembly level to the most detailed surface level.

Baxter (Baxter, 1994) presents the creation of a functional data model to represent the assembly information in a Product Model. In this approach, Baxter mentions several requirements that must be accomplished:

- *Support the use of rule based applications*
- *Model function inputs and outputs*
- *Support structural and functional viewpoints*
- *Reflect the origins of a product's function structure*
- *Incorporation within a central Product Model*

The function data model developed represent the relationships between functions and the product requirements they meet. The functions proposed by Baxter for represent assemblies in a functional way are: Attach, contain, control, drive, fix, isolate, locate, lubricate, restrict, seal, support, store energy, transfer energy, transfer material, transform energy, transform material, visual signal. These functions were used in an express representation of a meta-structure representation. The product specification was defined as a list of product requirements and product specifications.

Baxter's work tackled the aspects of assembly and leaves for further research the inclusion of the representation of material, geometric and functional requirements.

In the PSL project (Schlenoff, *et al.*, 1996), mentioned in section 2.3.3, the assembly topic has not yet explicitly studied as one of the main requirements for modelling processes. At present, it is only considered as an input for Production Scheduling: *"Assembly tasks would be a strong candidate for state representation since the actual process of assembling the parts is not of primary importance"* (Schlenoff, *et al.*, 1996). The reason because this project doesn't consider assembly is because it is more focused in the actual machining processes and the activities related to them, such as process planning or production scheduling.

## 2.5. Concluding Remarks

The literature review reported in this chapter performed a survey on the main areas related to this research.

The influence of assembly in the cost of a product is very high, if assembly considerations could be taken into account in the early stages of the product development, the cost of changes would be minimised and their effect maximised. Design For Assembly methodologies are traditionally targeted to the final stages of the product design, by performing assemblability evaluations. Assembly Process Planning is still in its infancy, while Assembly Sequence Generation has been broadly studied using several different approaches.

Although DFA and APP share a considerable amount of information for performing their tasks, there are few researches that consider the possibility of integrating them and make them work in an integrated approach. The use of information models could help the integration of such applications.

Product and Manufacturing Models have proved to be an efficient way for storing information about the product and the facilities of an enterprise. These information models can provide information to support product design from a concurrent engineering perspective. The approach of using a Product Model is broadly extended and the Manufacturing Model is gaining acceptance in the research community.

There have been some research that used Product and Manufacturing Models to capture information about machining and plastic injection processes. In spite of the importance of assembly in the cost of a product, there has been little research in the application of information models to support assembly related activities.

The issues found were used in the definition of the scope for this research, and are reviewed in chapter 4, where the contribution of this research in the context of the problem area is highlighted.

# 3. RESEARCH ENVIRONMENT

## 3.1. Introduction

This chapter presents the research environment in which this work was carried out, and introduces the major concepts that served as basis for this research.

The convenience of using information models to support the product development cycle is recognised by the research community. However, improved methods are still needed to assist the developer in the definition of information model structures. Currently available methods and standards can only help in certain stages of the information modelling process. This chapter presents the combination of tools and methodologies applied for defining the structure of information models that support applications through the product development cycle.

## 3.2. Research Environment and the Manufacturing Information Models Project

The work reported in this thesis was done in close relationship to the Manufacturing Information Models (MIM) research project (EPSRC GR/L41493), done in the Department of Manufacturing Engineering of Loughborough University.

The MIM project had as an antecedent the MOSES project (Model Oriented Simultaneous Engineering System), that was realised in collaboration between Leeds University and Loughborough University. That research focused on a computer based system to provide product and manufacturing information through the use of a Product Model and a Manufacturing Model linked by an integrated environment to a number of applications. The system enabled decision support based on these information models and was able to support Concurrent Engineering (Molina *et al.*, 1995).

One of the objectives of the MIM project was to generate new understanding about the roles of Product and Manufacturing Models and the enhanced data structures required to support the generation of manufacturing information in order to meet the needs of global manufacturing. Particular reference was made to the role of operations/process planning tools and NC part programming.

The MIM project followed two paths; the first related to machining activities, taking into account design for manufacturability and post-design activities that led to process planning (Zhao, 1999), (Cheung, 2000). The second path, investigated in this research, the support for assembly related activities. In the MOSES project support for assembly related activities was considered for the Product Model, but not fully developed nor implemented (Henson, 1995a), as mentioned in section 2.4.6.1.

In the MIM project, the approach of using a Product Model and a Manufacturing Model was followed; this research is also based on that approach. This work focused on enhancing the models by introducing structures for supporting assembly activities and by exploring the way in which to deal with highly product-dependant information.

The results of the research were tested with real enterprise information provided by industrial collaborators. For the assembly-related work of the project, a company that produces large electrical machines provided information about their products and facilities. Some of this information has been used to populate the information models, as explained in chapter 9.

The broad concept explored in this research is shown in Figure 3.1. The Product and Manufacturing Models are shown in the top part of the diagram and the application software to be supported with those information models is located at the bottom. The Integration Environment shown linking the Information Models and the data-driven applications is required to enable these elements to work together, even if them are based on different platforms or are located in different sites. This integration environment has to satisfy the requirements of the different elements, so that the models may store and maintain the information and the applications perform particular functions and access the information [Harding, 1996 #202]. The structure of the information models and the interactions between them are explained in chapter 7.

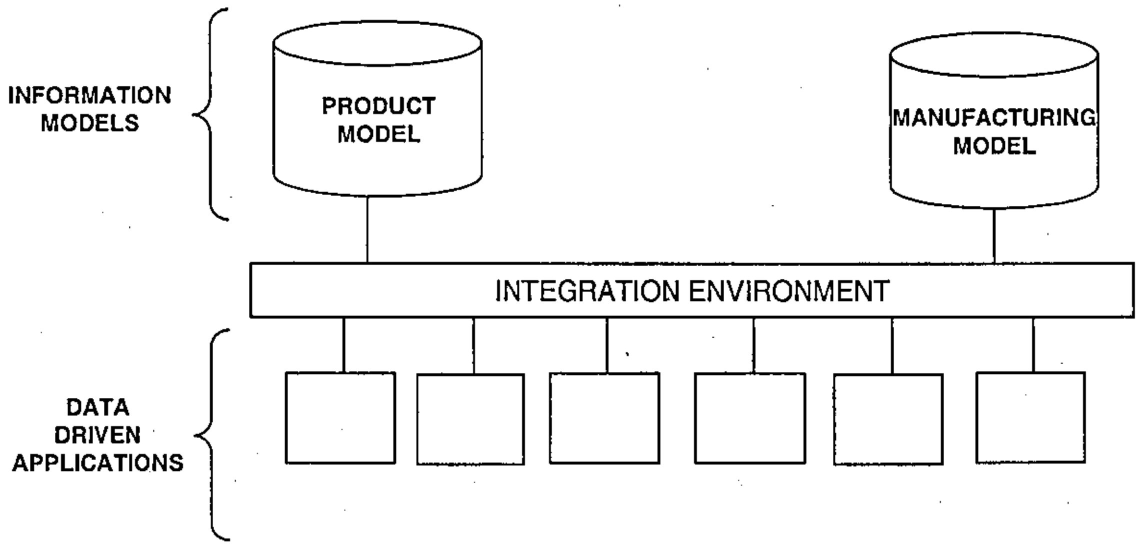



Figure 3.1 Structure of the approach followed in the research

## 3.3. Application of Methodologies in the Exploration of Information Models

In order to capture all the requirements for supporting the product development team, within the information models, some methodologies must be followed. Some of the approaches used in the research community provide the framework and background for understanding the information needs of the enterprise and the way in which the information must be structured, while others are tools designed specifically to capture and model part of the information. Molina (Molina, 1995) classified them in terms of:

1. Data/Information Modelling Methods: Data Flow Diagrams, Entity Relationship Diagrams, IDEF1x, NIAM and EXPRESS
2. Processes Modelling Methods: IDEF0, SAMM, IDEF3
3. Behaviour Modelling Methods: IDEF2 and Petri Nets
4. Hybrid Modelling Methods: Semantic Nets, Conceptual Graphs and Object-Oriented Methods.

While all the methodologies that are currently used in information modelling have some advantages, they also have weaknesses. In this research the combined use of IDEF0, IDEF3 and UML methodologies were used in the context of the RM-ODP standard (Reference Model for Open Distributed Processing), ISO 10746.

The RM-ODP provides a framework for designing coherent flexible distributed systems. The architecture of the RM-ODP provides a standard upon which a wide range of specifications for distributed systems can be integrated (Toh, 1999). Because of the complexity of modelling large amounts of information and capturing all aspects of information, the RM-ODP is divided into five viewpoints that express concepts and rules relevant to a particular area, for the description of a computational system. These viewpoints are enterprise, information, computational, technological and engineering, as shown in Figure 3.2. The functions that the information system will perform are defined in the enterprise view. In the information view, the information is described in terms of the information structures and flows. The other three viewpoints are for software implementation support. (Molina, 1995). This standard is used as a reference in this work.

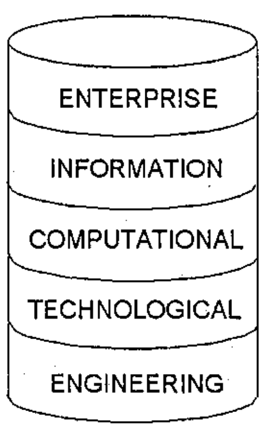



Figure 3.2 View levels according to the RMODP standard

### 3.3.1. Object Oriented Methodology Used In The Research

This section gives an insight to IDEF0, IDEF3 and UML methods that were used to define and create the structure for the information models and the application software. A more complete description of the methods has been included as appendix A.

#### 3.3.1.1. IDEF0

IDEF0 is a technique for modelling the functions (activities, actions, processes, operations) required by a system or enterprise, and the functional relationships and data (information or objects) that support the integration of those functions.

This modelling technique is based on combined graphics and text that are presented in an organised and systematic way to gain understanding, support analysis, provide logic for potential changes, specify requirements, or support systems level design and integration activities (NIST, 1993).

An IDEF0 model is composed of a hierarchical series of diagrams that display increasing levels of detail describing functions and their interfaces within the context of a system (NIST, 1993). The two primary modelling components are functions (represented on a diagram by boxes) and the data and objects that inter-relate those functions (represented by arrows), as shown in Figure 3.3.

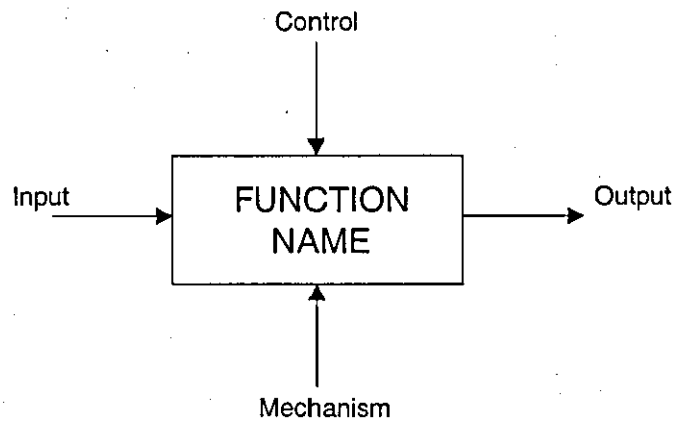



Figure 3.3 Basic IDEF0 representation of an activity and its related information

A set of IDEF0 diagrams can exist as either generic 'as is' or 'as should be' models (Colquhoun and Baines, 1991). The 'as is' model allows to evaluate the present situation of the system and the 'as should be' helps to define the strategies to follow in order to improve the system by describing the information flow necessary to support each activity.

IDEF0 is widely used in the research community due to its flexibility and clarity for modelling activities and the information flows between them. It is particularly good for evaluating enterprise activity structures as it provides an easy to understand model which non-experts can understand and assess. Although IDEF0 is good at providing an initial view of activity decomposition, it is incapable of modelling information process flows. This is due to its lack of time dependency input. The representation of information in IDEF0 is also weak as it is captured purely by simple textual description. These limitations impose a strong restriction on the use of this model for the design of information systems, so the need for other methodology able to capture the sequences of processes and information structure is evident.

In section 5.2, the IDEF-0 representation of the assembly activities in the product life cycle is given.

#### 3.3.1.2. IDEF3

The IDEF3 Process Description Capture Method was created specifically to capture descriptions of sequences of activities. It is therefore, a good option to use it as a method after an IDEF0 model has been created as the activities modelled in IDEF0 can be used as a basis for the definition of the processes. The IDEF3 has been extensively used for modelling processes, aiming to provide a general-purpose description of them. Among the main advantages of this methodology are its simplicity and descriptive power (Huang and Kusiak, 1998), (Kusiak and Zakarian, 1996), (Zuobao, *et al.*, 1996).

An IDEF3 Process Description is created using two knowledge acquisition strategies: a *process-centred* schematic and an *object-centred* schematic.

In IDEF3 process-centred diagrams, boxes represent types of happenings, as depicted in Figure 3.4. The term *units of behaviour* (UOB) refer to such happenings. Each UOB represents a real-world process. The arrows connecting the boxes indicate the precedence relationships that exist between the processes being described. Junctions represent constraints of the activation logic for the process. The junctions are used to represent different conditions in which an activity is carried out, as if they were logical operators: *and, or,* and *exclusive or* (Mayer, *et al.*, 1995).

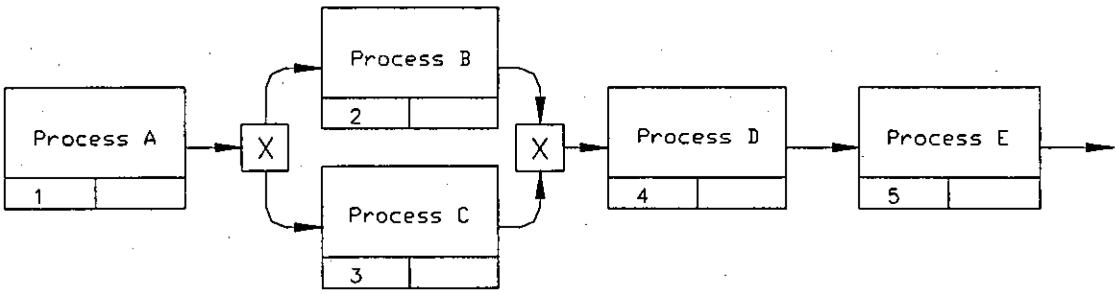



Figure 3.4 IDEF3 Unit of Behaviour boxes linked by relationships and junctions

The purpose of Object Schematic is to identify the possible states in which an object can exist. Though a real-world object often evolves through a continuum of states, an Object Schematic focuses on those distinguished states of particular interest to the domain expert. In Figure 3.5 three states of an object are shown, the boxes that are included between the stages are known as *referents,* and are included to enhance understanding and provide additional meaning to the process. They represent the processes that occur between the stages of the analysed object. They are also a link between the process and object schematics.

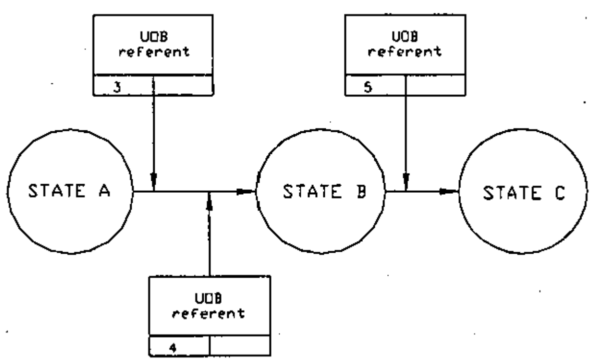



Figure 3.5 IDEF3 Object schematic description, showing an object in three stages with Unit of Behaviour referent boxes

IDEF3 also includes a set of Elaboration Forms to aid the process of capturing information. In these forms, the name of the UOB is captured together with the objects involved in the process, facts, constraints and description.

The notation and format in which IDEF3 models present the information is very useful for communicating with collaborators or future users of the system under development. On the other hand, the format is still very descriptive and unstructured. It is therefore inadequate as a basis for building software structures. So, other methods have to be used to identify the classes with their attributes and behaviour, which constitute the object-oriented structure of the software.

The IDEF-3 process and object centred description diagrams of the assembly activities are explained in section 5.3.

#### 3.3.1.3. UML

Unified Modelling Language is the result of the merging of three modelling methods, Booch, OMT and Objectory, taking some concepts from other methods. UML is a language used to specify, visualise and document the artefacts of an object-oriented system under development (Quatrany, 1998).

The elements that are used in the construction of the models are:

- Structural elements: Class, interface, collaboration, use case, active class, component, node
- Behavioural elements: Interaction, state machine
- Grouping elements: Package, subsystem

The relationships between elements can be dependencies, associations and generalisations.

In terms of the views of a model, UML defines several diagrams, including Use Case Diagrams for capturing the system functionality, Class Diagrams, and various behaviour diagrams for capturing the dynamic behaviour. These diagrams provide multiple perspectives of the system under analysis or development. The underlying model integrates these perspectives so that a self-consistent system can be analysed and built.

The Unified Modelling Language (UML) is a standard modelling language that must be applied in the context of a process. The UML authors propose a process named Rational Unified Method, explained in (Quatrany, 1998). Texel and Williams (Texel and Williams, 1997) propose other approach, which is more detailed in the steps that should be followed in the creation of a system with Use Cases.

The Rational Unified Method is an iterative and incremental process that evolves continuously into the final system. The iterations consist of one or more of the following process components: requirements capture (what the system should do), analysis and design (how the system will be realised), implementation (production of the code), and test (verification of the system). All of these analysed through the following phases: inception (specifying the project vision), elaboration (planning the activities), construction and transition (supplying the product to the user).

UML captures and documents decisions made during the inception and elaboration phases of system development (Quatrany, 1998). In the inception phase, UML uses Use Case Diagrams, in which the relationships between actors and use cases are represented. The actors represent anyone or anything that must interact with the system. Use Cases represent the functionality of the system, there are two kind of relationships between them, *uses* and *extends*.

In the elaboration and construction phases, the Class Diagrams and State Transition diagrams are used. A class is a description of a group of objects with common properties, common behaviour, common relationships with other objects, and common semantics.

The classes are represented as rectangles in which the name of the class is defined together with their attributes and operations. The relationships with other classes are represented using lines with adornments signifying associations, aggregation, or other relationships, as shown in Figure 3.6.

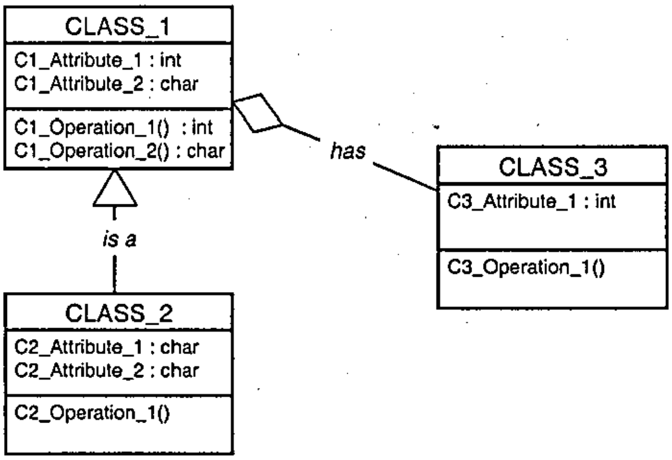



Figure 3.6 Basic structure of the class diagrams

State transition diagrams represent the states of a single object, the events or messages that cause a transition from one state to another, and the actions that result from a state change.

### 3.3.1.4. Combined IDEF and UML Methods

In order to define the structure for the information models, there are basically two approaches that can be followed. The first is a top-down approach, which starts with the analysis of the Enterprise level in order to obtain information about the general requirements and the policies that orient the design of the software system. Once the high-level is complete, the Information and Computational levels have to be built, in order to specify the structure of the system. The second method is a bottom-up approach, in which the process starts with the definition of the software functionality requirements from the user's point of view. After the computational level is defined using those functionality requirements, the information and enterprise levels are created.

In this research, a combination of both approaches was followed. For the top-down approach, IDEF0 and IDEF3 were used to determine Enterprise and Information levels, as shown in Figure 3.7. IDEF0 is commonly used, with some success, to model enterprise activities and information flows. However, its ability to describe information is weak and it cannot model process flows. IDEF3 offers both a process flow capability that can be linked to IDEF0 and the ability to model information in object centred descriptions. Furthermore, both methodologies have been proved to be very good means of communication with the collaborating companies.

The results obtained with the IDEF3 method provided key information for the creation of class attributes and operations that can be used in the design of computational systems using UML. The resulting UML class diagrams show the relationships and inheritances that are the main input to the creation of object-oriented databases that hold the data of the information models.

However, those diagrams were not totally complete, since the point of view of the system user was not included in the analysis. So, it was necessary to apply the bottom-up approach, starting the analysis with the definition of the application software requirements. In order to do that, Use Cases were defined for the system. The resulting information allowed the completion of attributes and operations for the classes defined. The complete Class Diagrams were used to build the structure for the databases that constitute the Information Models.

These combined methods and approaches have been used to model the information required through different stages of the product life cycle for the assembly of large electrical machines. The IDEF0 and IDEF3 methods were applied for obtaining the process centred descriptions of the collaborating company and of a generic design process including DFA and APP, as detailed in section 5.2. UML Class diagrams and Use Cases were subsequently used to construct the computational view of the system. The testing software system was designed and developed using object-oriented concepts. The information models obtained are discussed in section 5.3.

The use of the described combined methodology has shown to provide an improved definition of the relationships between the stages of information model definition. A published journal paper (Dorador and Young, 2000) about this combined methodology is included as appendix B.

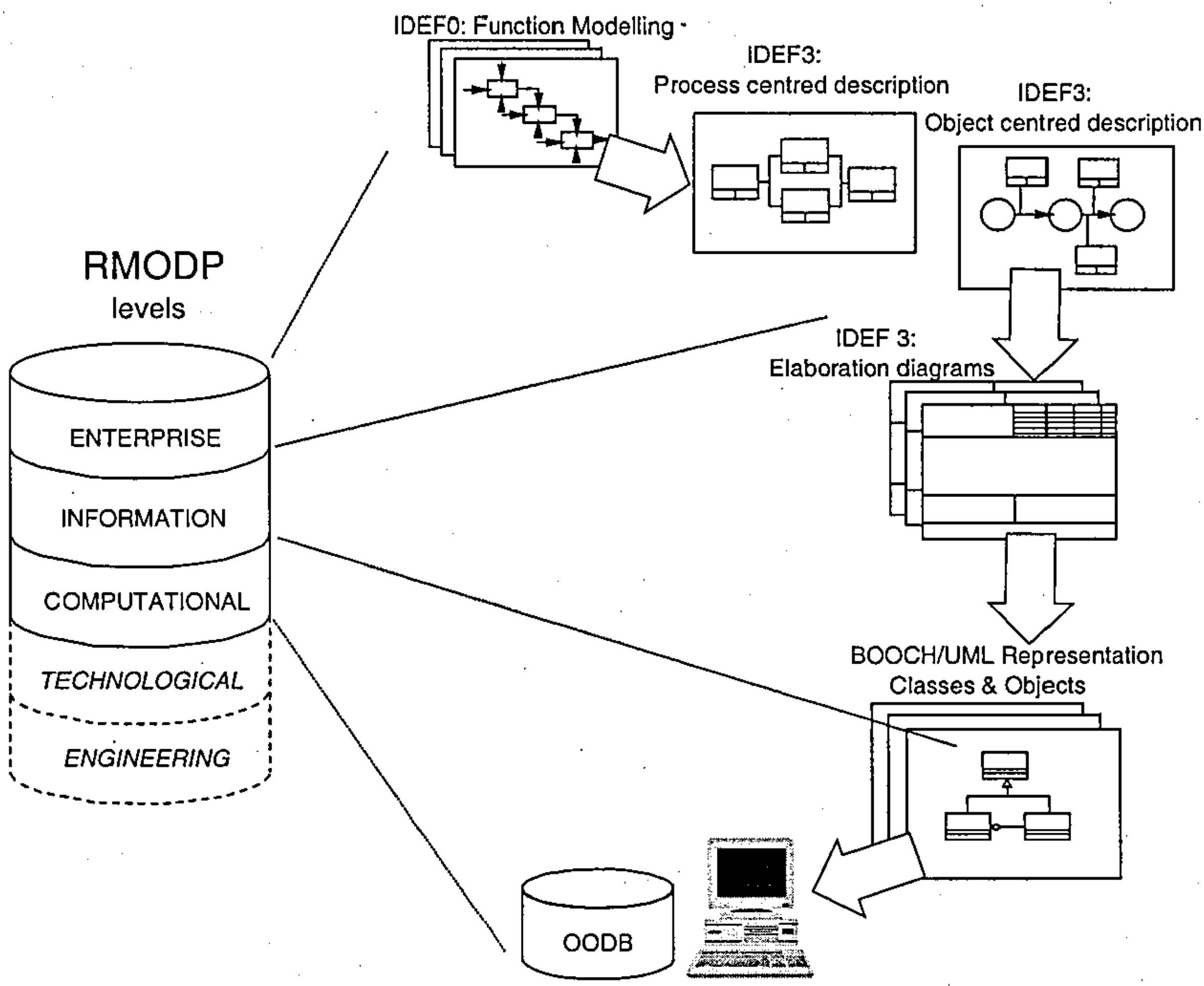



Figure 3.7 Methodologies be used to design the information systems (Dorador and Young, 1999)

### 3.3.2. Experimental Environment

#### 3.3.2.1. Microsoft VISUAL C++ ©

The software for populating the Information Models and the DFA and APP application software was created using Microsoft Visual C++. This software allows the development and implementation of visual interfaces between the user and the system.

The functionality of the software application has been realised by using Microsoft Visual C++©, as a software development environment. This tool allows the development and implementation of visual interfaces between the user and the information models stored in the database. The tool uses an object-oriented approach and is based on windows MFC (Microsoft Foundation Classes) (Pappas and Murray III 1997).

#### 3.3.2.2. OBJECTSTORE ©

ObjectStore is an object-oriented database management system that allows to create and modify C++ objects, instead of tables, columns, rows, and tuples (sets of ordered data separated by commas) used by relational databases. ObjectStore allows describing, storing, and querying complex data used in sophisticated computer applications, as well as data traditionally managed by relational database applications. It also allows to persistently storing data independently of the data type (ObjectDesign, 1998). ObjectStore was used to implement the information model schemas of the Product Model and Manufacturing Model, this implementation is detailed in chapter 5.

ObjectStore has two tools that simplify the creation and visualization of the object-oriented databases, namely the ObjectStore Database Designer and ObjectStore Inspector.

ObjectStore Database Designer uses UML or OMT class diagrams as input to generate the C++ code for each class in the database schema. The classes generated define the information structure (persistent data) of the information models mentioned above, i.e. Product Model and Manufacturing Model.

The ObjectStore Inspector is a tool that allows visualization of the object's data and relationships, in the populated database file.

## 3.4. Summary

Moving from a business level description of the system to a formally described structure for the information is something that current modelling methods do not provide. The use of the combined methods described in this chapter allowed modelling the different levels, from the enterprise level to the computational level.

An important part of the research environment was the interaction with the collaborating company, which provided information for testing the research ideas.

The use of the mentioned methods allowed a logical approach for the definition of the structures for the information models. In particular, the following points can be raised:

- IDEF0 has proven to be a very good start point to understand the enterprise and the functions that are performed in it through the product life cycle.
- The IDEF3 process centred description diagrams can be constructed with information from the IDEF0 and going in more depth in the areas that are relevant to the research.
- The IDEF3 object centred description allows to track the changes that an object suffers through the product life cycle and to identify the actors that perform those changes.
- In the IDEF3 the "Units of Behaviour" document the information of the facts and constraints that exist in every state of the object and of the processes involved. This information is useful in the creation of the object-oriented information models.

- The UML (Unified Modelling Language) representation is very useful to exchange ideas and show graphically the classes that are necessary in the creation of the information models and of the applications that will use them. The use of the UML representation helps to define the attributes and operations contained in the classes. Those diagrams also allow the representation of the interactions between classes through "is-a" and "has" relationships, depicting the concepts of object-oriented approaches.

# 4. INFORMATION STRATEGIES IN ASSEMBLY DECISION SUPPORT SYSTEMS

## 4.1. Introduction

This chapter argues the research contribution of this thesis, describes the issues relevant to the problem area, and provides an outline of how these problems have been addressed.

The issues on assembly decisions support systems are explained in the first section of this chapter. Assembly information model structures defined for supporting a product development team through the early stages of the product life cycle are explained in the second section of the chapter.

The proposed method for the integration of Design For Assembly and Assembly Process Planning through assembly information modelling is introduced in the last section of this chapter.

## 4.2. Assembly Decision Support Systems

Concurrent Engineering has proved to be a valuable approach for answering some of the demands and pressures that the market imposes on industry. One of the main characteristics of this approach is the intensive information interchange between different areas that interact through the product development cycle (Pels, 1996), (Lim *et al.*, 1995), (Prasad, 1996).

Decision support systems include approaches that use algorithms, agents, knowledge-based systems and information modelling. This thesis contributes in the area of decision support systems using information modelling, by providing computational support to the development team.

# 4. INFORMATION STRATEGIES IN ASSEMBLY DECISION SUPPORT SYSTEMS

## 4.1. Introduction

This chapter argues the research contribution of this thesis, describes the issues relevant to the problem area, and provides an outline of how these problems have been addressed.

The issues on assembly decisions support systems are explained in the first section of this chapter. Assembly information model structures defined for supporting a product development team through the early stages of the product life cycle are explained in the second section of the chapter.

The proposed method for the integration of Design For Assembly and Assembly Process Planning through assembly information modelling is introduced in the last section of this chapter.

## 4.2. Assembly Decision Support Systems

Concurrent Engineering has proved to be a valuable approach for answering some of the demands and pressures that the market imposes on industry. One of the main characteristics of this approach is the intensive information interchange between different areas that interact through the product development cycle (Pels, 1996), (Lim *et al.*, 1995), (Prasad, 1996).

Decision support systems include approaches that use algorithms, agents, knowledge-based systems and information modelling. This thesis contributes on the area of decision support systems using information modelling, by providing computational support to the development team.

By modelling the information and making it available to different applications, problems of lost data, lack of flexibility, and misinterpreted information can be solved. The use of a Product Model is extended through the research community and has proved to be useful (Young, 1992), (Krause *et al.*, 1993), (Van Der Net *et al.*, 1996), (Xue, 1999). An additional model to hold the information of the manufacturing facilities is gaining acceptance in the research community (Molina, 1995), (Pels, 1996), (MANDATE, 1998), (Young, 1999).

The need for an integrated environment in which the information could be interchanged between different applications to support product development through the lifecycle, particularly for assembly-related activities, was identified by several researchers (Sturges and Kilani, 1992a), (Rampersad, 1995c), (Grewal, 1997). Some approaches followed for creating integrated environments for assembly are explained in section 2.4.5.7. This thesis provides the definition and design of integrated information structures for Product and Manufacturing Models to support assembly-related activities, particularly for the integration of Design For Assembly and Assembly Process Planning software applications.

Product and Manufacturing Models have been applied successfully in several areas, such as machining (Cheung, 2000), plastic injection (Young, 1999), and reverse engineering (Borja, 1997). However, only a limited amount of work has been done to support assembly-related activities.

The need for information structures across product assembly and assembly processes and systems has been identified by several authors (Baxter, 1994), (Henson, 1995b), (Regli, 1997). Some work was done in the definition of Product assembly information structures in the STEP standard and in the MOSES project, as mentioned in chapters 2.3.5.7 and 2.3.7.

While formal DFA and APP methods have proven to be useful in the detailed design stage of product development, these are currently not applied in the early stages, where they could make a greater impact in the product's cost. This is mainly because of the lack of specifications and detailed information.

The need for methods to support assembly in the early stages of product design has been raised by several authors (Mantyla, 1990), (Zha, *et al.*, 1999a), this thesis contributes to this area by providing a system able to support developer's decisions from the early stages of product design. This system helps the development team by providing guidelines for assemblability and the capabilities of the resources and processes that will be used for performing the assembly operations. In the stages in which the specifications of the product are not completely defined, the system uses information from the Product and Manufacturing Models to help the development process. This is explained in chapter 5.

## 4.3. Information Model Structures

This thesis defines Information Structures to support assembly in the different stages of product development. Section 5.2 presents an analysis of the way in which product development activities can be supported with the Information Models defined. This includes and emphasises the support given in the early stages of development. The use of Product and Manufacturing models allows the support for selection of resources and processes. In order to test the usability of the mentioned information models, some rules of DFA and APP were implemented, as explained in section 4.4.4.

The broad concept explored in the research is depicted in Figure 4.1. In it the cylinders inside the Product Model and the Manufacturing Model represent the information structure related to assembly, defined in order to support the Assembly-related activities during the product life cycle. The Design for Assembly (DFA) and the Assembly Process Planning (APP) software applications are illustrated with an arrow between them representing the interactions explored in the research in order to allow the work in a concurrent engineering environment. This interaction allows the designer to take into account the main APP considerations when applying the DFA during the design stages of the product life cycle in contrast to the traditionally independent application of DFA and APP, although they require common information.

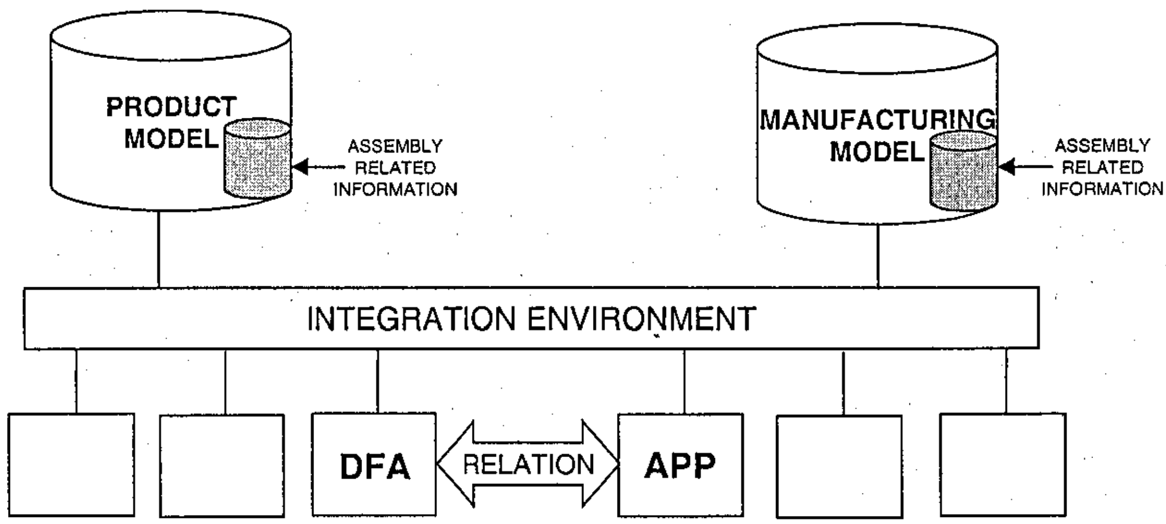



Figure 4.1 Information Models and Assembly application interactions (Adapted from Dorador and Young, 1999)

Henson (Henson, 1993) identified the importance of supporting assembly using a Product Model and proposed the integration of behaviour and structure based on the product's physical structure, and the inclusion of mating conditions in the model (Henson, 1995a). The structure of the Product Model defined in this thesis includes the functional and physical characteristics of the product, as well as the mating conditions.

Several researchers (Molloy *et al.*, 1991), (Lim *et al.*, 1995), (Hsu, 1993), (Taylor, 1997) have identified the need of having a way for allowing the interactions between DFA and APP, and different solutions have been proposed, as explained in section 2.4.4. One of the issues that remains unresolved by those approaches is the one of having a general way for sharing the information between DFA, APP and other software applications in order to allow concurrency. This thesis explains how to define assembly information structures that allow the transparent sharing of information among different software applications.

The starting point for this research was the Manufacturing Model structure proposed in the MOSES project, and some aspects of the MOSES' Product Model and the work done in STEP for assembly representation. This is depicted in the structures shown in Figure 4.2.

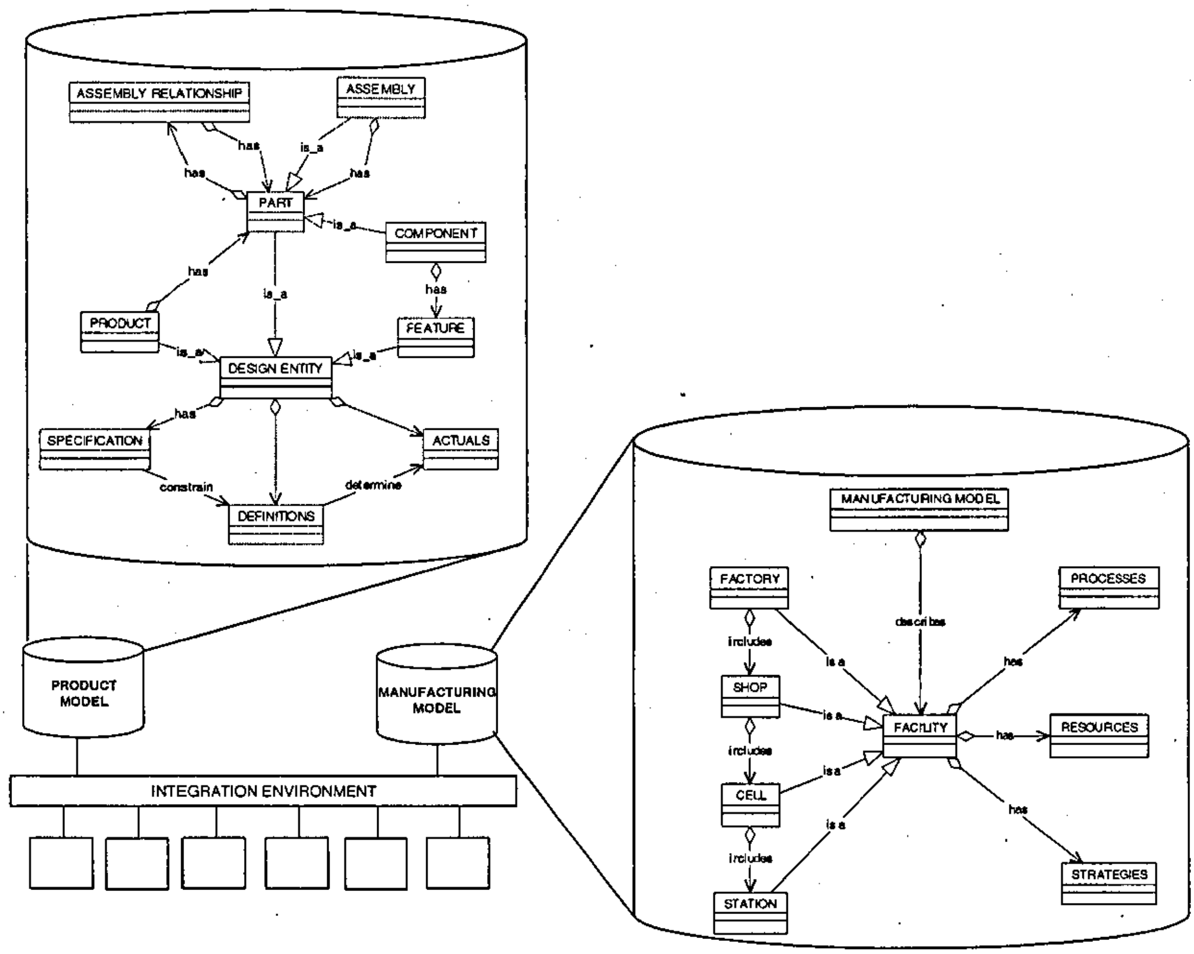



Figure 4.2 Starting point for the research, adapted from (Molina, 1995) and (Borja, 1997).

In order to define structures of the Product Model and Manufacturing Model that allow supporting assembly-related activities, it was necessary to explore the information requirements through the product development cycle and also the procedures followed in an enterprise. On the one hand, there are several methodologies that help to capture and understand the information about activities and processes and in the other, there are methods for defining structures for databases based on user requirements. However, there is a lack of an integrated methodology that allows the modelling of the information from the capture of the activities and requirements of an enterprise and translates it directly into classes and objects required for building an object-oriented database. Although the creation of a methodology was out of the scope of this thesis, IDEF standards and UML methods were applied and proved to be useful for defining the structures of the information models. This is explained in section 5.2 and a published journal paper about the application of these methods is included as appendix B.

Three general issues to solve in this thesis are related to:

- Product Model structure
- Manufacturing Model structure
- Interactions between the Product and the Manufacturing Models

These issues are explored in the following sections.

## 4.4. Assembly Information Model Structures

For Information Models to be effective, the structures underlying are critical because they dictate which information is kept where and how to catch the information necessary to provide support to the software applications. The Product and Manufacturing Models defined serve as source and repository for the product and facilities information. The Assembly related information is stored in those Information Models, while the assembly functionality exists in the data driven applications.

### 4.4.1. Product Model Structure

The Product Model structure proposed in the MOSES project was used as a starting point, as shown in Figure 4.2. Even though in that structure Assembly is considered as the link between parts, an issue to be solved was the clarification of the assembly levels that could exist in a product (parent assembly, subassembly and component). In the Product Model defined in this thesis, three interrelated classes were created, namely Parent, Subassembly and Component, all of them are members of the Product class, as shown in Figure 4.3. So, the products can be instantiated in these three levels, in order to capture product families and, in a further work, allow the reuse of the design information.

Another issue was the inclusion of a way to represent variant, original or standard parts, in order to provide the facility to use the information of existing products for the creation of design variants. This was explored by creating a *Product Range* class that contains the products with their associated characteristics. This is depicted in Figure 4.3.

A basic requirement of a Product Model is the clear description of the product. In this thesis, such description is based on its characteristics, which include specification, geometry, functionality, manufacturing information, rules, material properties, etc.

The Product Model defined for this thesis represents the product information to support the applications that are interacting in the product's life cycle. The Product Model structure must allow those applications to have access to the information and also to be a repository for the data created by them.

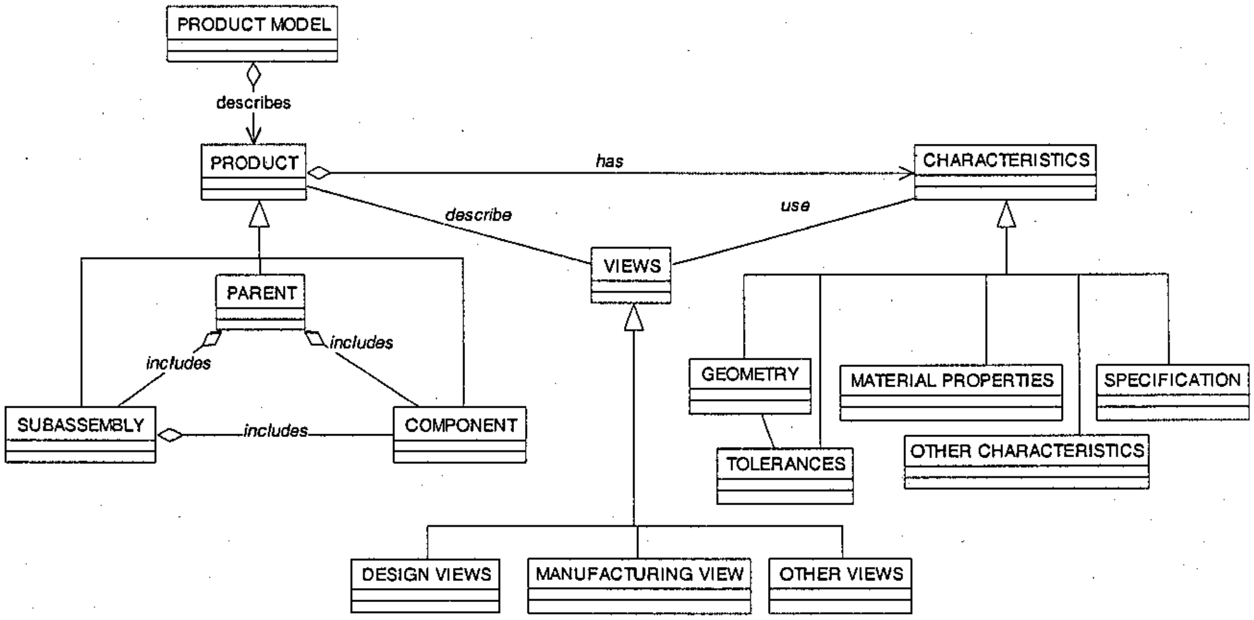



Figure 4.3 General structure of the Product Model using UML notation

A limitation of the current product models is that they present the information only from a particular viewpoint, and because different software programmes will use that information, it is convenient to be able to access the information from different perspectives. The author's work extended the structure of the Product Model to be able to support multiple views. The structure for presenting a design viewpoint and a manufacturing viewpoint, from the assembly perspective, were included in the Product Model structure. This allows the use of the information contained in the Product Model from a new perspective. This issue is explained in section 7.2.3.

### 4.4.2. Manufacturing Model Structure

The high level structure of the Manufacturing Model given on the MOSES project (Molina, 1995) was adopted as a departing point in the present thesis and explored within the context of assembly-related information. One of the main issues of the MIM project was to propose a way in which global activities could be supported.

The Manufacturing Model proposed in the MOSES project was focused on machining operations and did not consider assembly. That model included the definition of Resources, Processes and Strategies for representing a Factory, as explained in section 2.3.3. Several researchers have applied Resources and Processes for modelling machining operations. The issue of extending them for capturing assembly resources and processes was researched in this thesis. The exploration of required assembly information structures is explained in section 7.2, and depicted in Figure 4.4.

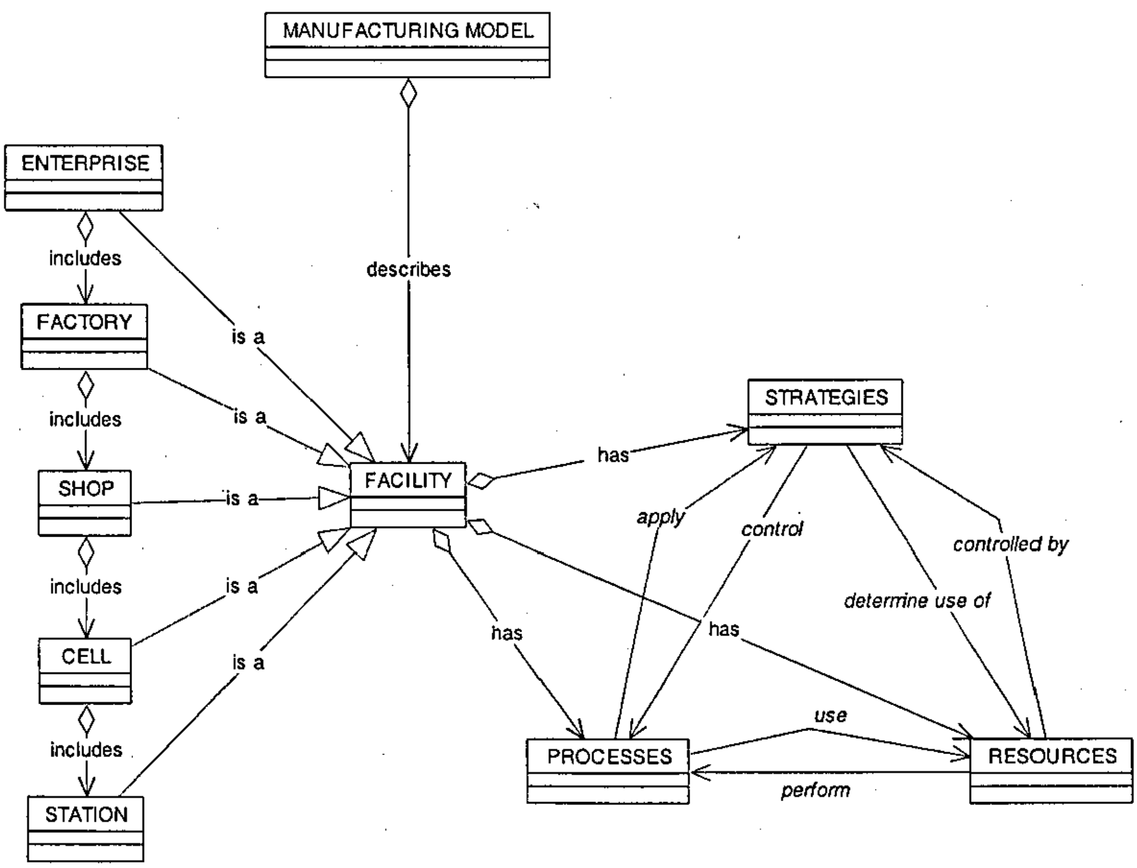



Figure 4.4 General structure of the Manufacturing Model using UML notation

Operational strategies were explored in the MOSES project, and the authors identified the need for further research to determine other kinds of strategies (Molina *et al.*, 1995). The research reported in this thesis explored strategies for performing assembly operations, tactical and strategic level strategies for selecting the kind of assembly system recommended, according to economic and technical parameters, and product family-related strategies. The latter are necessary because it has been identified that a Manufacturing Model for supporting assembly-related activities is highly product-dependant, and it needs to consider information contained in the Product Model. This is explained in detail in section 7.2.

In (Molina *et al.*, 1995) the need to study the interactions between Resources, Processes and Strategies in the Manufacturing Model is recognised. By holding Resources, Processes and Strategies in the same Model, it is possible to have a clear understanding of the relationships and dependencies between them. This thesis explores those interactions in order to allow a direct information interchange, as shown in Figure 4.4 and explained in section 7.3.

### 4.4.3. Interactions Between the Information Models

The information models proposed in the MOSES project (McKay, 1991), (Molina, 1995) are independent sources and repositories of information. Some researchers (Borja, 1997), (Costa, 2000) have included relations between the information models, the first created an application that used information from the Product Model and the Manufacturing Model in order to select factories using basic parameters of the product. The second performed interactions between a Product Model and a Product Range Model.

An important issue was to extend the possibilities of interactions between the Product Model and the Manufacturing Model in order to allow information models to support effectively a Concurrent Engineering environment. These interactions between the Information Models are depicted in Figure 4.5. This figure also illustrates the information interactions of the Information Models with the data driven software applications for Design for Assembly and Assembly Process Planning.

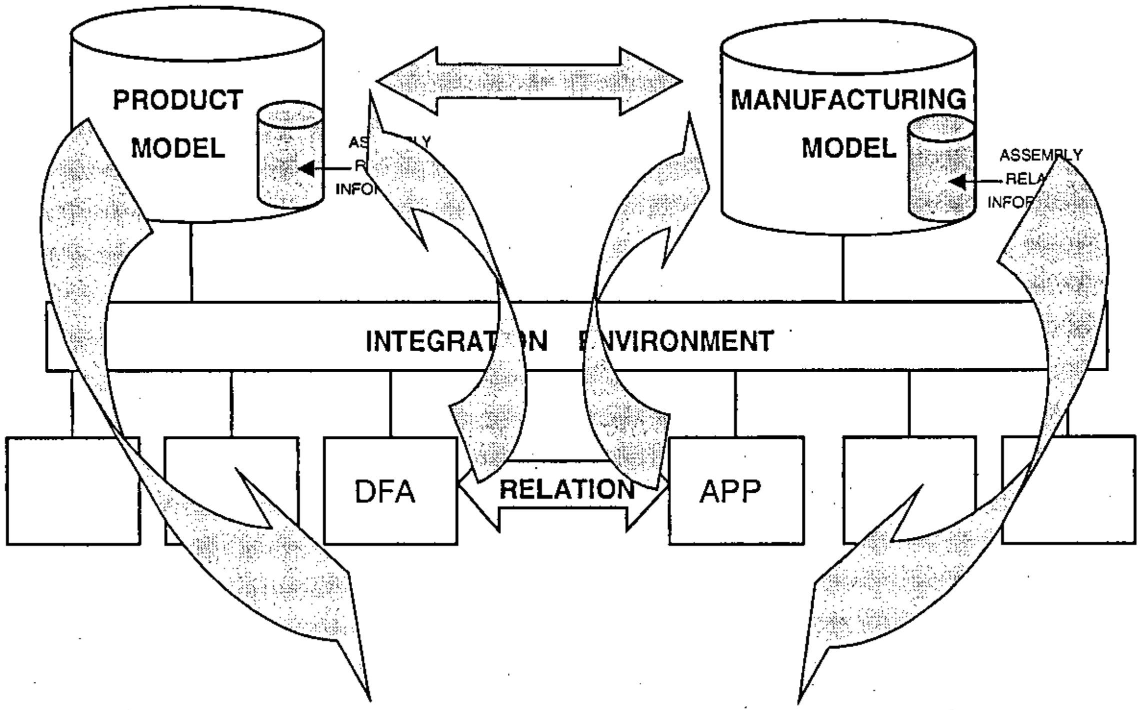



Figure 4.5 Interactions between Information Models and software applications

In this thesis an integral approach was followed in order to consider the effects on the complete system caused by any change in it. This approach allows the system to be a support for Concurrent Engineering, in which Product, Assembly Process, and Assembly System are considered simultaneously. When an aspect of any of them changes, everything changes, according to the relationship existing between them.

A significant issue in assembly information modelling, opposed to machining information modelling is that assembly strategies are heavily product-dependant. This dependency led to the proposal of an approach that requires an additional relationship between the models that incorporates the creation of product dependant strategies.

By creating product dependant strategies in the PM and then sending them to the MM, a new way of perceiving the MM is shown. Before, general strategies were created in the MM in order to show how the Resources and Processes could be used, so these strategies were independent of the characteristics of the product.

could be used, so these strategies were independent of the characteristics of the product.

In order to allow interactions between the Product Model and the Manufacturing Model, the first must have a structure that allows the storage of product-specific manufacturing information. Once this information exists, the following issue is to determine how to use it together with the Manufacturing Model. The results of the research about these issues are explained in section 7.2.

### 4.4.4. Integrating Design for Assembly and Assembly Process Planning through Assembly Information Models

As described in chapter 2, several researchers have proposed solutions to the issue of sharing information between Design For Assembly (DFA) and Assembly Process Planning (APP) due to the importance of both activities and to the similarities that the required information has. In this thesis, the above mentioned software applications are used to test the capability for supporting assembly-related application software with the information models proposed. Those applications, although simple, have the basic capabilities for providing the product development team with information and recommendations about the product and facilities. The functionality and interactions of these applications through the product development cycle is explored in chapter 6.

The support that DFA and APP can give in the different stages of product development was evaluated in the design of the structure of the Information Models used in this thesis. Supporting assembly in the early stages of product development is an issue that has been recognised by the research community (Hsu, 1996), (Simpson, 1995), (Nof, 1997). Sets of recommendations have been proposed in this thesis for the conceptual design stage. When information is introduced into the Information Model, DFA and APP check it and provide recommendations. In the embodiment design stage, a basic evaluation tool is used for checking the possible assembly problems that the individual parts or the assemblies have. Information about the resources, processes and strategies available for performing the assembly operations is given. The capabilities

Figure 4.6 represents the assembly-related applications supported through the concurrent development cycle by the assembly area proposed for the Product Model and Manufacturing Model. The overall representation given in this figure is not intended to represent time-based planning activities.

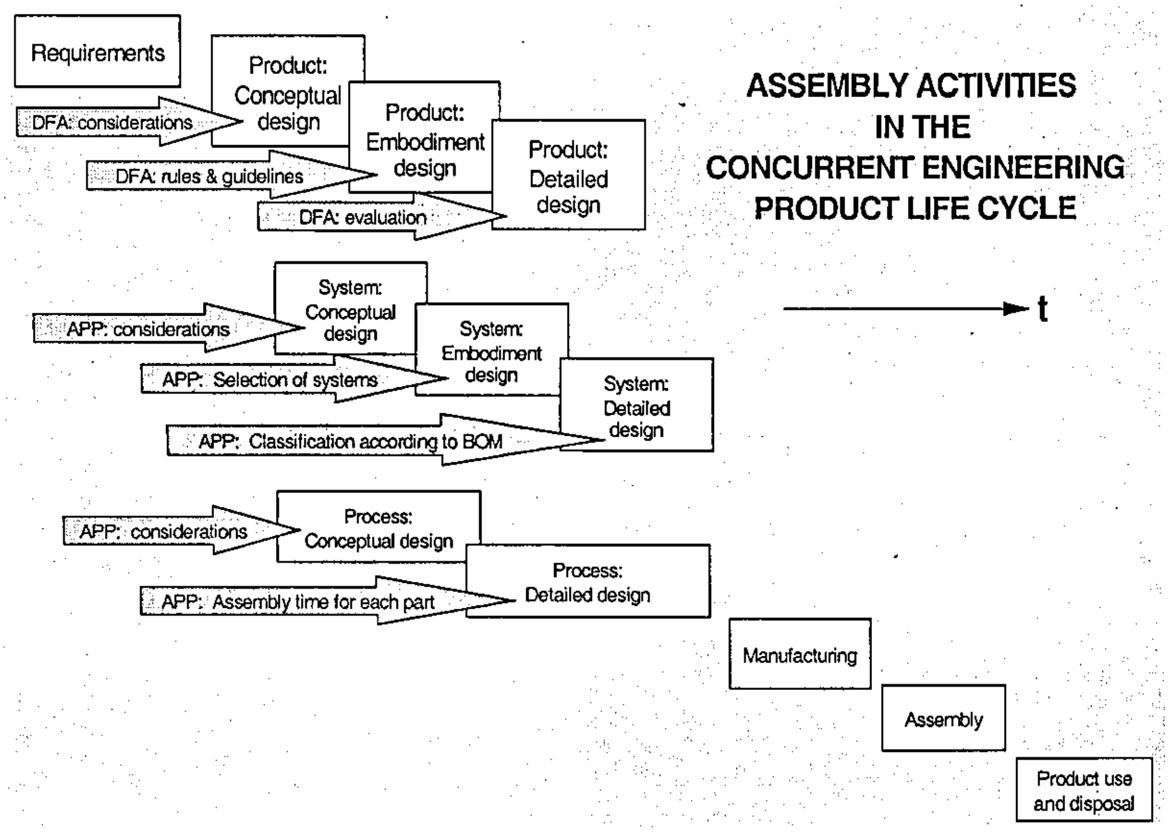



Figure 4.6 Stages of the product life cycle in which DFA and APP interact

## 4.5. Summary

This chapter presented an outline of the basic contributions made in this thesis, relating them to the issues identified in previous chapters, and situating these issues on the context of information modelling to support assembly-related activities.

The relevance of the system proposed in the thesis, which is based on integrated information models to hold the information about the product and the facilities available, was explained. This system provides a way for supporting assembly-related activities in a concurrent engineering environment through the product development cycle, particularly in the early stages of its development.

The structures for the Product and Manufacturing Models were introduced to support assembly-related activities, particularly the integration of Design For Assembly and Assembly Process Planning software applications.

In the following chapters of the thesis the contributions are explained and clarified. In Chapter 7 the structures of the Product Model and Manufacturing Model and the relations among them are discussed. Chapter 6 provides an insight into the relations between the DFA and APP application software. Chapters 8 & 9 explain the implementation and results of the tests performed with the testing software.

# 5. THE APPLICATION OF ACTIVITY, PROCESS AND INFORMATION MODELLING TO IDENTIFY THE GENERAL ASSEMBLY INFORMATION REQUIREMENTS

## 5.1. Introduction

This chapter identifies the requisites for the creation of the structures for the Product Model and the Manufacturing Model using the combined methodologies IDEF0, IDEF3 and UML. The basic consideration for the design of the structures was that they must allow the use of the models as a source and repository for the information required during the product development cycle by software applications running concurrently.

This chapter describes the process followed in order to define the general structures of the information models. The methodologies described in chapter 3.3 have been used.

IDEF0 and IDEF3 captured the understanding of the requirements imposed by the use of assembly-related activities, and by doing so, these modelling approaches set the information foundations from which the UML class diagrams were defined and the databases created.

In the top-down approach, the IDEF0 and IDEF3 methodologies were applied to produce models that help to capture the information about the processes, operations and objects that are involved in the product development cycle, and, in turn will need to be supported by the information models.

The complete set of IDEF0 and IDEF3 diagrams that capture the concurrent design process proposed by the author, focusing on DFA and APP, may be found in appendixes C and D.

## 5.2. Modelling Assembly Activities in the Product Life Cycle

There is the need of understanding the role of the assembly activities in the concurrent development cycle of the product, process and system. As explained in section 3.3, IDEF0 was chosen for obtaining that information.

An IDEF0 model of the functions and activities performed has been produced, with emphasis on assembly related activities. The complete set of diagrams is included in Appendix C. The scope of the major Concurrent Engineering Design activities of products, systems, and processes is as follows:

> The "design product" activities are related to the design of the product, from its definition in the conceptual design stage through to the finalisation of the design and transfer of the information to the production area.
>
> The "design process" function refers to the activities that must be performed in order to assemble the product.
>
> The "design system" function relates to the activities involved in the design and selection of the resources that will be used for performing the necessary activities for assembling the product.

The IDEF0 model for assembly interactions during the product life cycle was created with the consideration that the design of the product, process and system were performed concurrently, as shown in Figure 5.1. The intermediate results produced are taken into account when decisions are taken in other areas. In doing so, the benefits of working in a concurrent environment are realised. The complete product life cycle, from the analysis of the customer requirements through to the shipping of the product was modelled using IDEF0, emphasising the assembly-related activities. IDEF0 diagrams do not represent sequences of activities, these are captured using the IDEF3 method, as explained in section 5.3.

In order to analyse the interactions between them in a concurrent engineering environment and their relation with Design for Assembly and Assembly Process Planning, the product, system and process designs were divided in conceptual design, embodiment design and detail design stages.

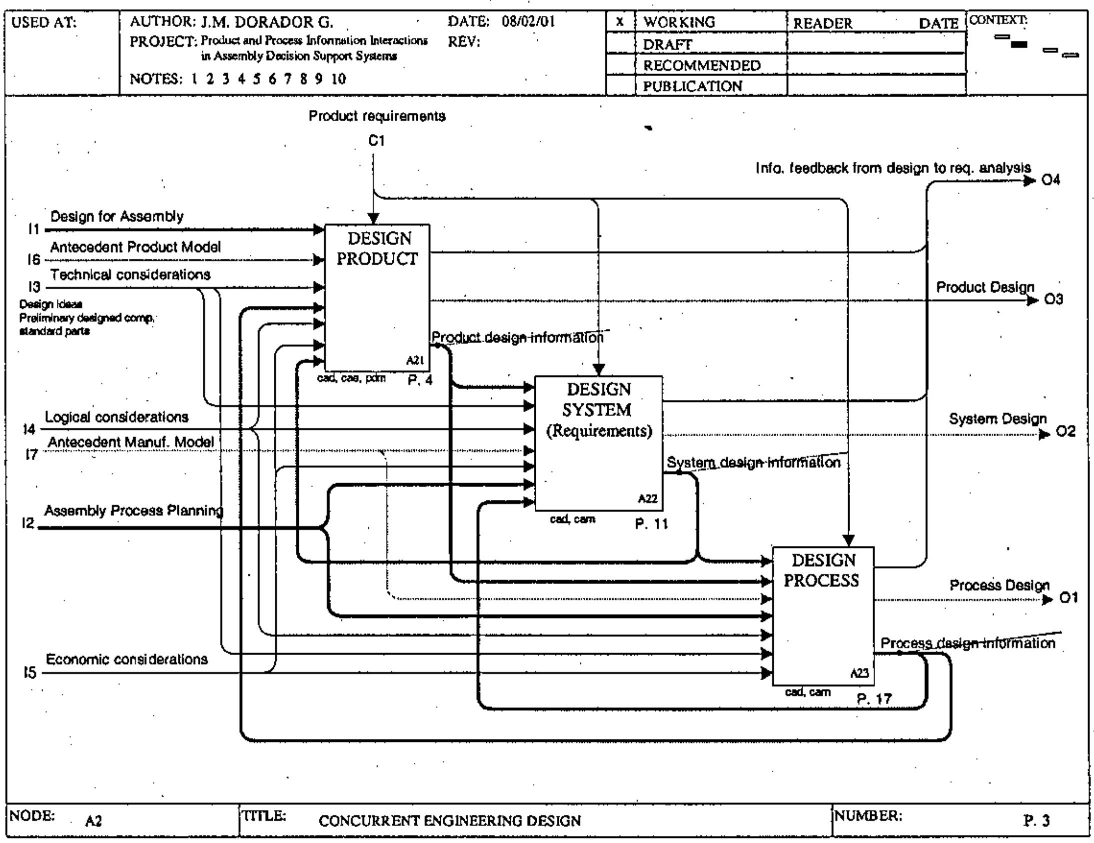



Figure 5.1 Representation of the Concurrent Engineering Design process

As mentioned before, the complete IDEF0 set of diagrams is included as Appendix C of this thesis. However, an example is given below to illustrate the way in which Design For Assembly and Assembly Process Planning influence the Concurrent Engineering process. Figure 5.2 shows the decomposition of the *propose configuration for main functions* of the Embodiment Design stage. The activity A21211 "Develop preliminary distributions" has as a control, namely the specification list produced in the Conceptual Design stage. The inputs for that activity are the *product design information, process design information* and *system design information*. The first is originated from the Conceptual Design stage and the other two reflect the concurrency of the model and come from the Process Design and System Design activities.

An additional input for activity A21211 was included to show the influence of Design for Assembly at this stage. The influence of DFA in this activity is by providing guidelines to help to create the division into subassemblies suitable for variant design, by considering reuse of the parts, interchangeability and modularity. The feasibility study of the proposed distributions uses criteria such as clearances for assembly/disassembly, parts integration, match of materials, and standardisation of parts and components.

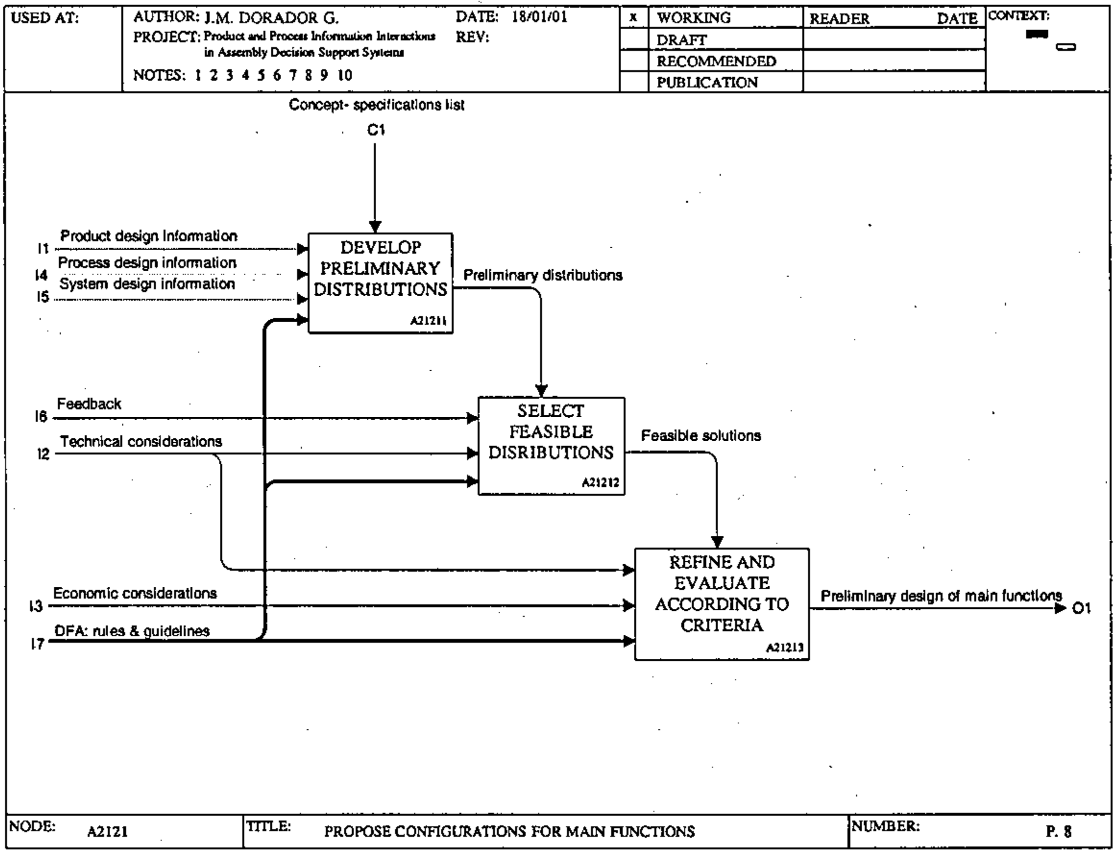



Figure 5.2 IDEF0 diagram for the *Propose configurations for main functions* activity

For the activity *Select Feasible Distributions*, the inputs are the feedback from the completion and optimisation of the configurations and technical factors that determine the feasibility of the proposed distributions. DFA considerations will interact within this activity by helping the evaluation of the possible distributions by applying assemblability criteria.

The last activity depicted in Figure 5.2 consists of the refinement and evaluation of the proposed distributions, where both the technical and the economical factors play an important role. The DFA rules and guidelines help by providing assembly-related criteria to evaluate and modify the selected options.

## 5.3. Modelling Assembly Processes

Once the phase of building the IDEF0 model for capturing the activities was completed, the IDEF3 methodology was used to capture process flows, which can not be modelled in IDEF0. IDEF3 methodology does not require models to be first produced in IDEF0, but when the system that is analysed is complex, it is proposed that IDEF0 models are first produced. The assembly interactions throughout the product lifecycle in a concurrent environment are highly complex; it was decided therefore, that IDEF0 was used for the activity modelling and IDEF3 used for the process and object centred descriptions.

Three independent IDEF3 process centred description models were developed; the first model is for the processes that take place in the design of the product, the second was of the system and the third was of the process. These models were constructed from the information captured in the IDEF0 model during the first phase of analysis.

The relation between design product, design system, and design process was established using IDEF3 process centred description diagrams. Concurrent engineering requires the relationships and interactions between the three design processes to be identified. This is represented in Figure 5.3 using lines between the processes.

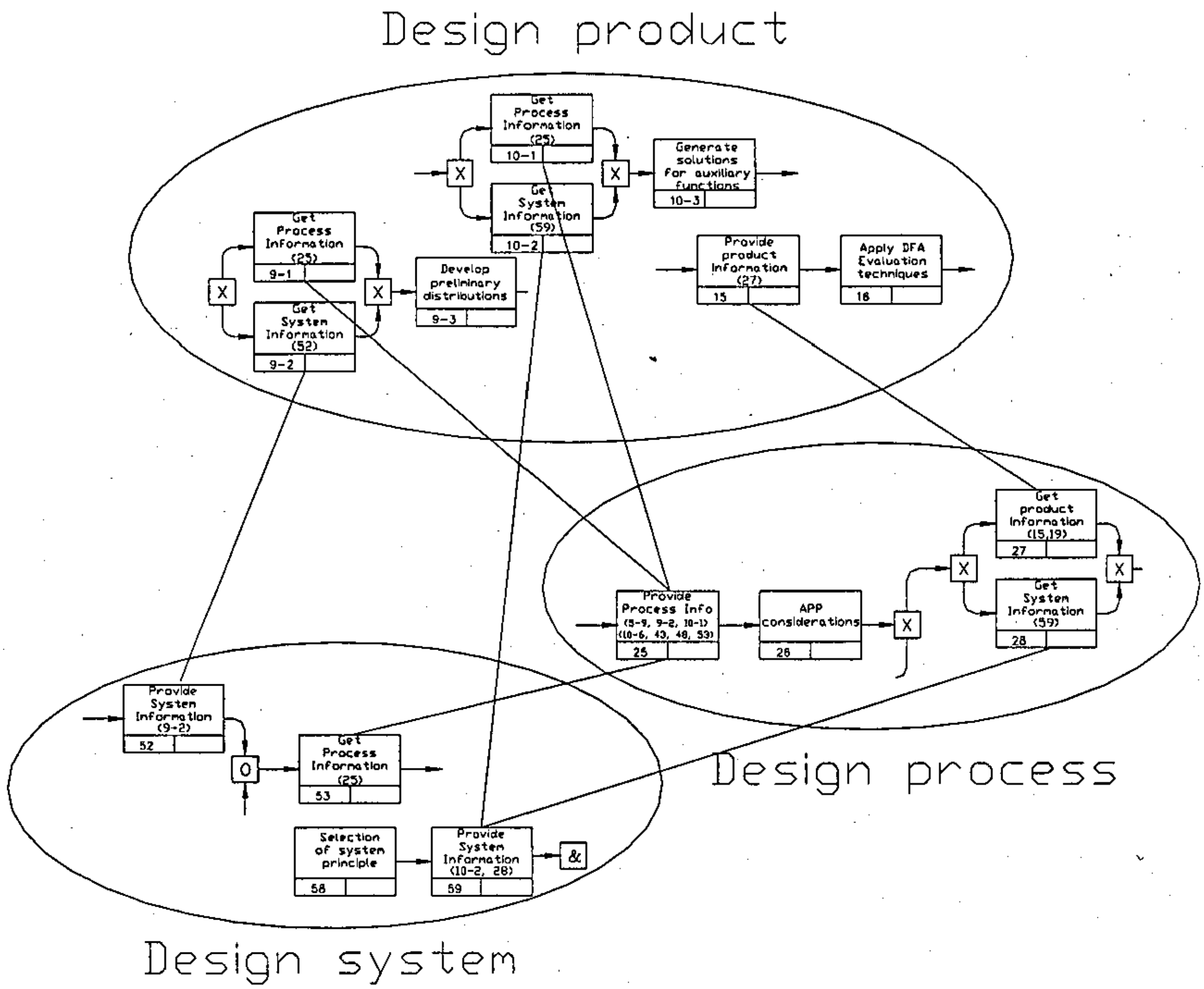



Figure 5.3 Example of relations using IDEF3 diagrams

The basic symbol of the Unit of Behaviour (UOB), has been modified by including a number inside a parenthesis in the activities that have relation with other models. The use of these numbers was decided due to the fact that this is a concurrent engineering process in which many interactions take place and the use of lines to link them would cause confusions.

The example shown in Figure 5.4 was taken from the product design model, and can be read as follows: UOB 10-6 (Get Process Information) obtains information from the UOB number 25, which belongs to the process design model. UOB 10-7 (Get System Information) gets information from the UOB 88, which belongs to the system design model. UOB 10-9 provides product information to the UOBs 60, 65 and 70 of the system design model.

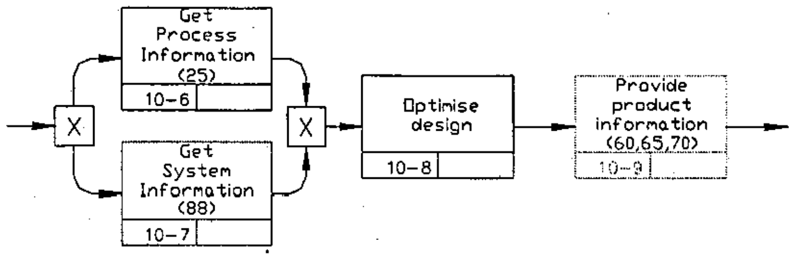



Figure 5.4 Detail of the references made by including a number in parenthesis

The IDEF3 models created for *design product, design system design process*, including the interactions with DFA and APP are included in Appendix D. A portion of the resulting diagrams is shown as an example in Figure 5.5. The process centred IDEF3 description diagrams proved to be very convenient for identifying the appropriate places to introduce the DFA and APP information to support processes.

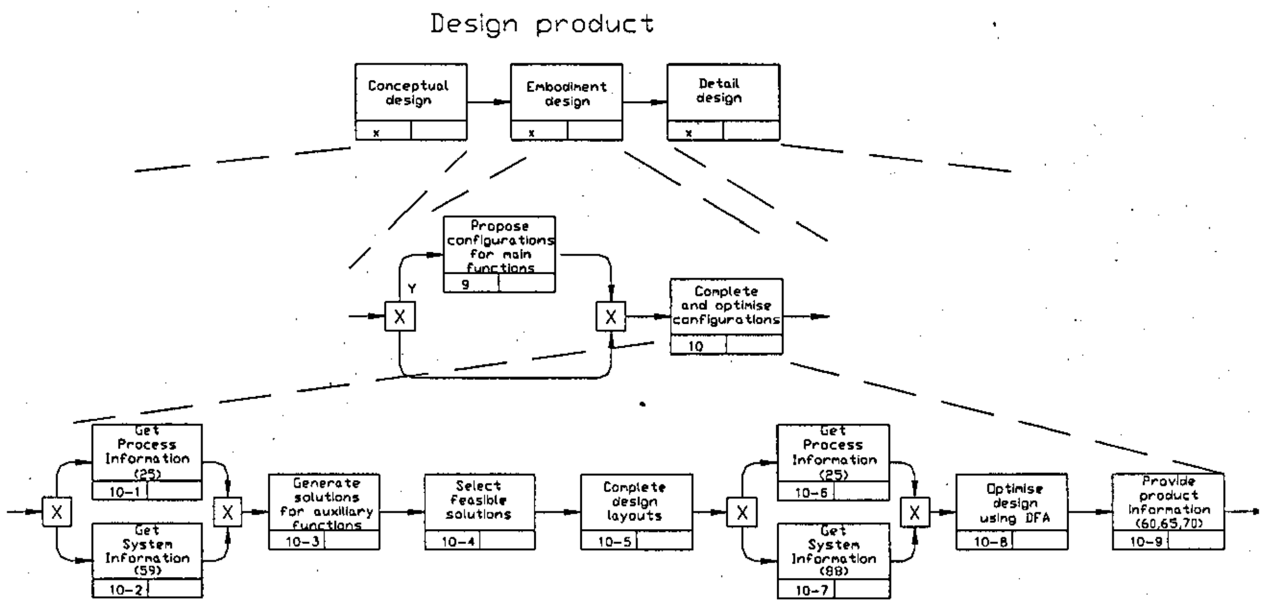



Figure 5.5 Part of the IDEF3 model for Design Product

Once the process-centred description was completed, it was used to create the object-centred description, which describes the changes that occur in an object throughout the product life cycle. The selected objects for the object-centred description were the product, the process and the system. This description gives a view of the transformations that take place during the different stages of product development.

In the IDEF3 object centred description some referents are shown between the different stages, these referents signify the processes during which the indicated transition occurs, or at least a process involved in the transition. Due to the focus of this research the selected referents for this model are those related to assembly topics.

The IDEF3 object centred descriptions were produced to display different stages in the evolution of the design of the product, process, and system. The diagrams illustrate that some transitions require several referents. The complete set of diagrams is included in Appendix D, and an example of these object-centred descriptions is shown in Figure 5.6.

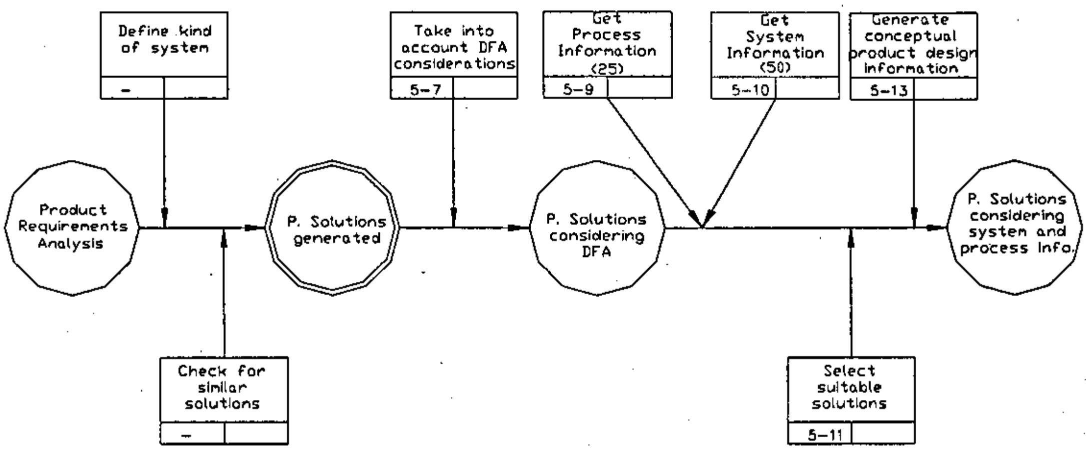



Figure 5.6 Extract of the object-centred description for the product design.

In Figure 5.7, an example of the sequence of the methods used is shown, using arrows to illustrate the translation of the information between models. The translation of information captured using IDEF0 to IDEF3 is not direct, there is the addition of the time parameter in the model. However, the elaboration of an IDEF3 process diagram starting with the information of the activities captured in IDEF0 has proven to be simple. In the IDEF3 object centred description, key objects have to be modelled in order to identify their interaction with processes. The elaboration forms are a good way for documenting the units of behaviour; considering that the facts and constraints captured will be useful for determining the attributes and behaviour of their related classes in the application of UML methodology.

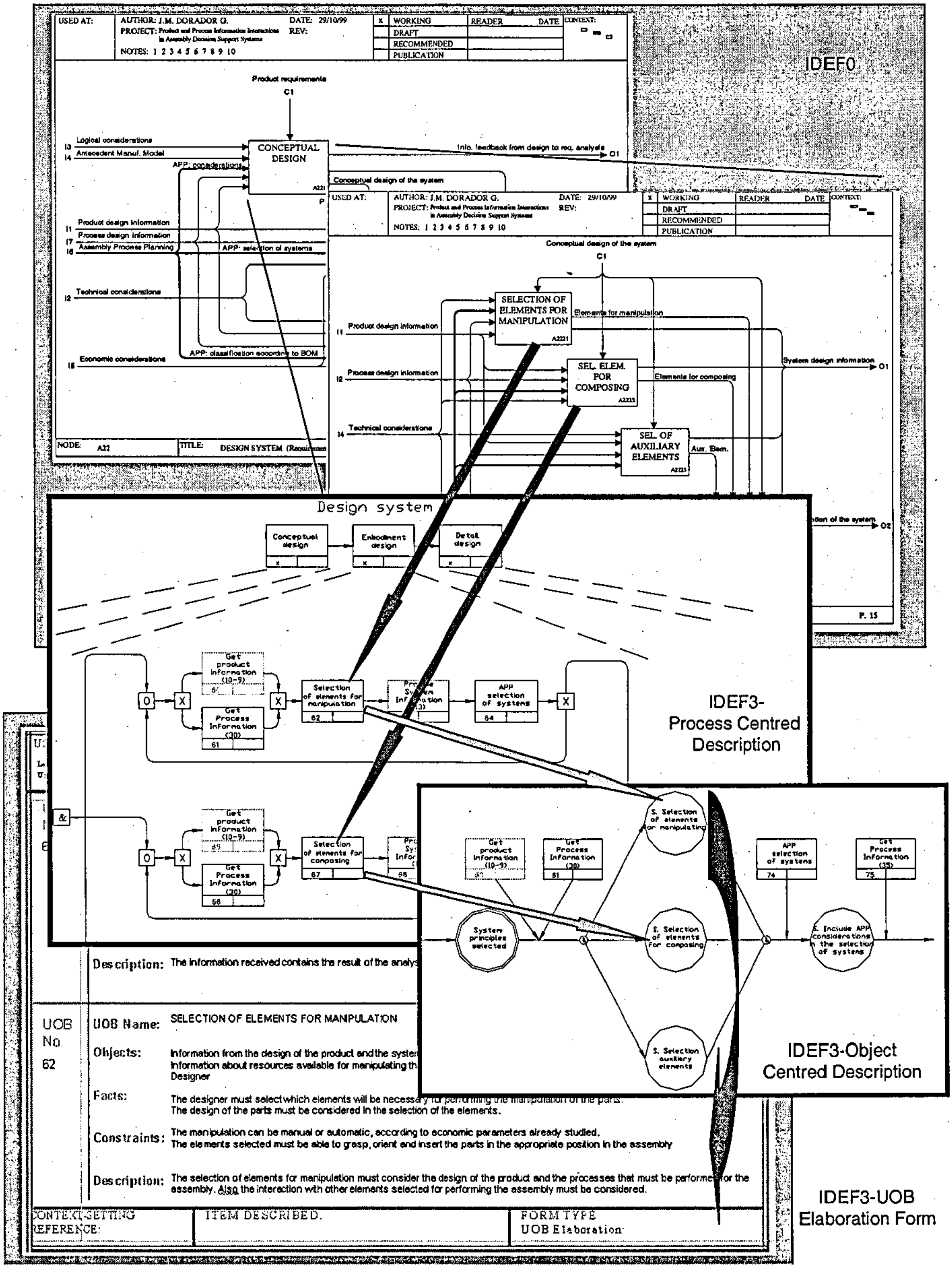



Figure 5.7 Relation between IDEF0 and IDEF3 methods

## 5.4. High Level Information Classes to Support Assembly

The methodology applied in this research for defining the structure for the information models was explained in section 3.3.1.4. The first step consisted of the application of IDEF0 and IDEF3 methodologies in a top-down methodology, in which the information required in the Enterprise and Information levels of the RM-ODP was obtained. The Computational level comprises the creation of the object-oriented description of the system and its implementation in the construction of the information models and software applications.

The IDEF0 and IDEF3 models created play a major role in capturing information about the processes, operations and objects that need to be supported by data from the Product Model and the Manufacturing Model. The UML methodology was applied to develop a description of the general structure of the information models and the requirements that the application software should fulfil.

Once the general structure of the information models was created, their detailed decomposition into classes and subclasses was based on the information contained in the IDEF3 diagrams and Unit of Behaviour charts. The overall approach is, therefore, a top down derivation of requirements using IDEF0 and IDEF3, and a bottom-up development of detailed information models that meets the requirements. The development of the detailed information structures based on general structure of the information models offers a further advantage in that the models offer potential re-uses.

As explained before, the first stage of the Rational Unified Method is the requirements capture, this can be done with the application of Use Case Diagrams, this models the system considering the external users of the system (Actors). However, this is done assuming that the system designer has developed a good understanding of the requirements for the software. The information captured by the IDEF0 and IDEF3 models plays a crucial role in providing this understanding.

In the construction of the UML Class Diagram, the relationships between classes have to be defined; these relations can be of aggregation or generalisation. This information can be obtained from the Unit of Behaviour elaboration diagrams obtained in IDEF3. At this stage, the top-down and bottom-up approaches meet. An example of the use of an IDEF3 elaboration diagram in defining the classes is shown in Figure 5.8. The general class structures defined for the Product and the Manufacturing Models were depicted in figures 4.3 and 4.4.

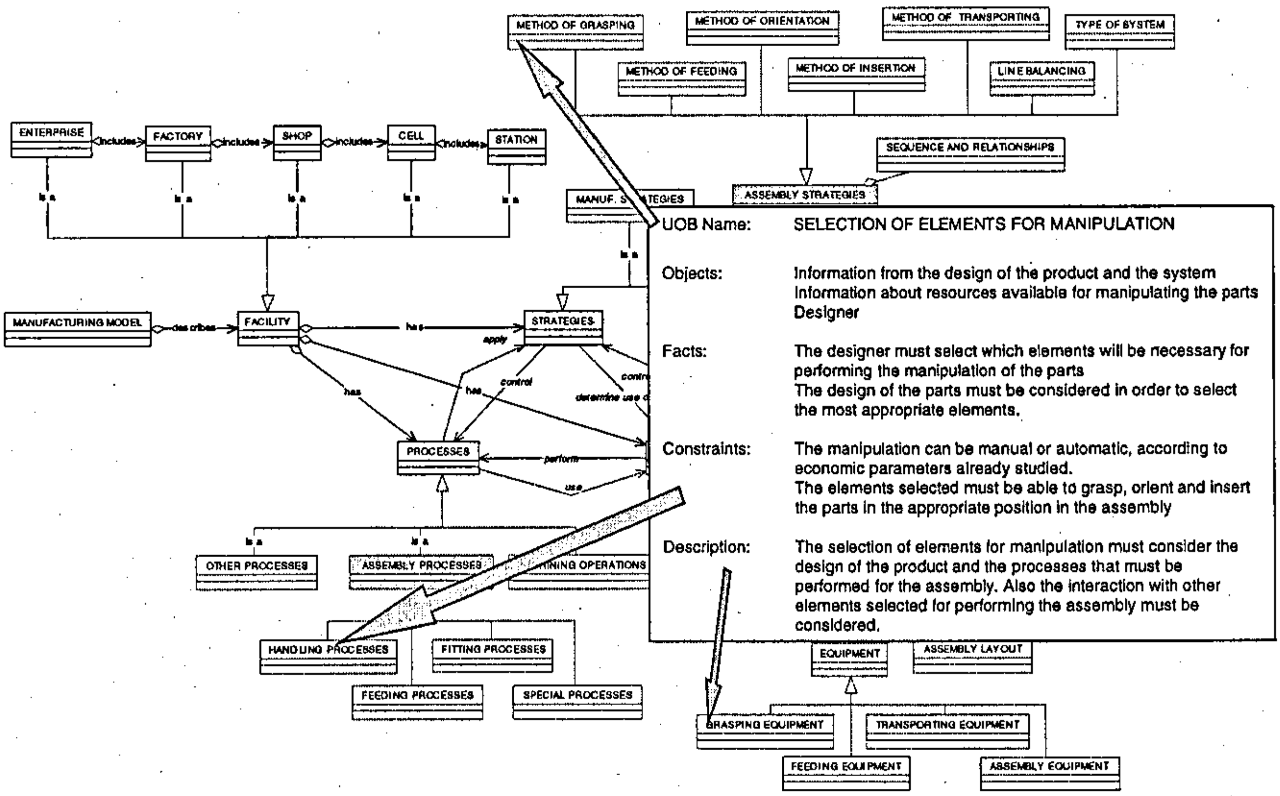



Figure 5.8 Use of IDEF3 UOB Elaboration Diagrams in the definition of subclasses in the UML Class Diagram of the Manufacturing Model

Once the basic system's functionality has been established, and the functions that will be supported are defined, the following step is to define the basic structure of the information models. The requirements that were identified in the IDEF models are included in the structure. It is necessary to include the attributes and behaviour in the classes, a section of the Manufacturing Model class diagram is shown in Figure 5.9. The complete class diagrams for both information models are included and explained in chapter 7.

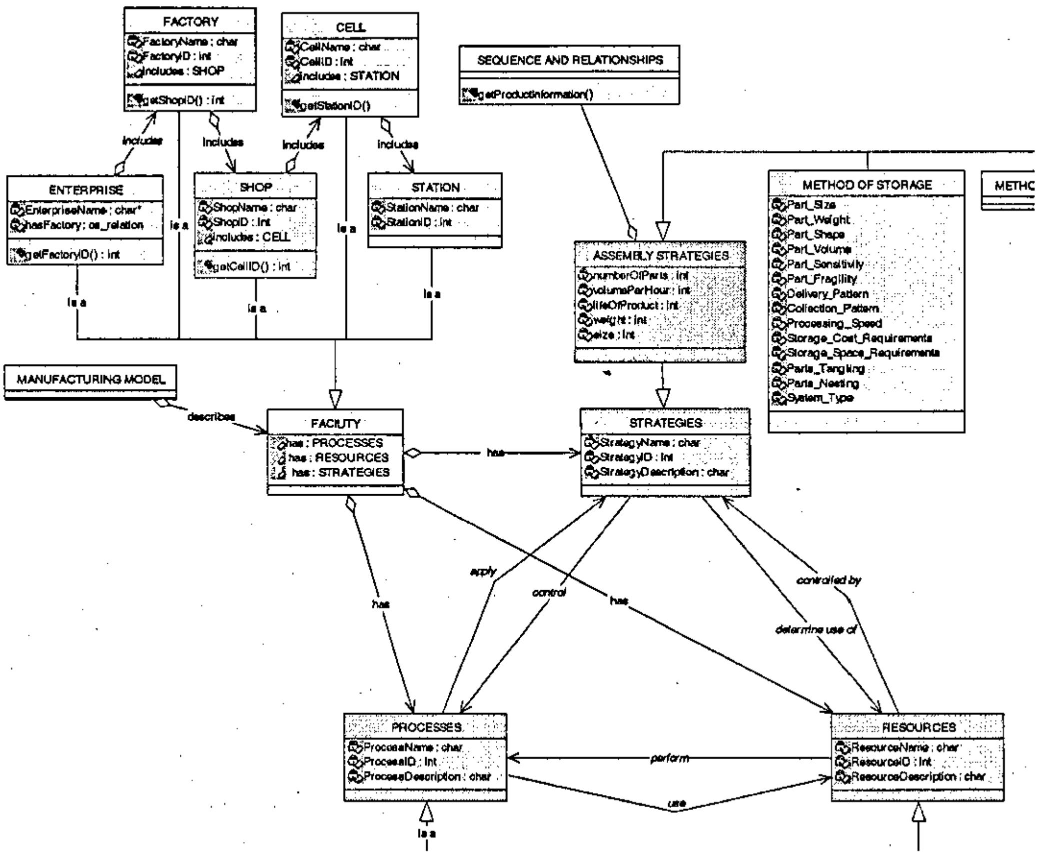



Figure 5.9 Section of the UML Class diagram for the Manufacturing Model

## 5.5. Summary

The methodology applied for defining the structures of the information models proved to be very useful. IDEF0 and IDEF3 proved to be very effective in modelling the information at the enterprise and information levels in compliance with the viewpoint requirements of the RM-ODP. The combined use of these methods overcomes their individual limitations.

The UML class diagrams gave the necessary details to the general structure of information, detailing classes, defining their attributes and operations, and building the relationships with other classes. It has been found that the UML methodology needs extra support in the "requirements capture" stage, mainly for the definition of the structure of the information models. Although there is no direct interaction between the information modelled in IDEF3 and the UML methodology, the IDEF3 modelled information proved to be very useful as a guide in the definition of the classes' attributes and behaviour.

Modelling the concurrent design process with IDEF0 and IDEF3 provided a clear view of the stages in which DFA and APP interact with the design process, besides the clear representation of the concurrency in the design of the product, the process and the system (resources). The above mentioned interactions between DFA, APP, and the information models is described in chapter 6, and the definition of the DFA and APP application software is included in chapter 8.

# 6. THE EVALUATION OF DESIGN FOR ASSEMBLY AND ASSEMBLY PROCESS PLANNING TO IDENTIFY ASSEMBLY INFORMATION REQUIREMENTS

## 6.1. Introduction

This chapter explains how the functionality of Design for Assembly (DFA) and Assembly Process Planning (APP) were evaluated to identify the information requirements through the product, process and system development cycle.

General information structures require specific information about the applications that they support in order to include the appropriate attributes in their classes. The general structures for supporting assembly-related activities, given in chapter 5, require specific information about the specific requirements of DFA and APP.

This chapter evaluates the influence of DFA and APP and their information requirements through the concurrent development cycle. The IDEF3 diagrams explained in chapter 5 are extended, and in order to obtain the information requirements, the Elaboration Diagrams for the Units Of Behaviour were done, these diagrams are included in Appendix D.

Section 2.4 explained several approaches that the research community has followed in the study of DFA and APP. After analysing those approaches, a collection of the core functions for DFA and APP was identified and it is shown in Table 6-1. These functions provide the starting point used to define the information requirements that DFA and APP impose on the Information Models that support them through the development cycle.

| Functions of Design for Assembly | Functions of Assembly Process Planning |
|---|---|
| • Identify characteristics that affect assembly<br>• Consider functionality of the part<br>• Identify assembly sequence<br>• Insertion analysis<br>• Evaluate joining methods<br>• Assess number of parts and products to assemble<br>• Recognise feeding problems<br>• Recognise manipulation problems | • Produce a precedence diagram<br>• Assess time required<br>• Identify tools for manipulating parts<br>• Identify resources available and select equipment<br>• Identify assembly sequence<br>• Create Bill of Materials<br>• Create work sheets<br>• Support line balancing |

Table 6-1General functionality for DFA and APP

The integrated approach presented in this thesis has the potential to improve the ways in which DFA and APP are performed because of the use of Information Models to support them. With this support, DFA and APP can be used effectively at the early stages of the development cycle in a concurrent engineering environment, and not only applied at the detailed design stage, as is traditionally done due to the lack of information in the early stages of design.

## 6.2. Design For Assembly Application Functionality

DFA activities have influence throughout the design process. In the IDEF-3 model created for the product design, shown in section 5.2.2, key DFA activities can be identified throughout the different stages of the design process. These are analysed in this chapter extending the above-mentioned IDEF3 description. Special emphasis is made on the support that DFA provide in the early stages of design, where the more influential decisions can be made. Those early stages are evaluated in sections 6.2.2 and 6.2.3, while section 6.2.4 evaluates the detailed design stage. Section 6.2.5 provides the set of information requirements that DFA imposes on the Information Models. This section starts by evaluating the requirements needed for determining the kind of system to be used.

### 6.2.1. Determination of the kind of system that will be used for Assembling the Product

The kind of system used for assembling the product determines how to apply DFA techniques, because is not the same analysing a product that will be assembled manually or one that will be assembled by a dedicated automatic machine. The product development team has to determine the ideal system for assembling the products. Assembly systems can be manual, automatic with specialised machines, flexible in cells or lines using robots, or a combination of these. In order to decide the kind of system, it is necessary to estimate production volume, life of the product, size, weight, complexity, and number of the parts.

The classifications of assembly systems proposed by (Lotter, 1986), (Andreasen, 1988), (Boothroyd *et al.*, 1994), (Boubekri and Nagaraj, 1993) have been used. Table 6-2 contains guidelines for the selection of systems.

| Volume/hr | Life of products (years) | Number of parts | TYPE OF SYSTEM | |
|---|---|---|---|---|
| <100 | | | Weight < 2.5 kg, size < 30cm | Assembly Station |
| | | | Weight 2.5–15 kg, size 30-90 cm | Modular Assembly |
| | | | Weight > 15 kg, Size > 90 cm | Custom Assembly |
| 100-250 | ≤ 2 | | Multistation assembly | |
| | ≥ 3 | 2-8 | Multistation assembly | |
| | | > 8 | Assembly cell | |
| 250-500 | ≤ 2 | | Multistation assembly | |
| | ≥ 3 | 2 | Multistation assembly | |
| | | 3-4 | Assembly cell | |
| | | > 4 | Dedicated line | |
| 500-1000 | ≤ 2 | | Assembly cell | |
| | ≥ 3 | | Dedicated line | |
| >1000 | ≤ 2 | | Assembly line | |
| | ≥ 3 | | Dedicated automatic assembly machine | |

Table 6-2 Guidelines for the selection of assembly systems

In addition to this table, Boothroyd (Boothroyd *et al.*, 1994) proposes that if there is insufficient information at the beginning of the development process to determine the kind of system that will be used, an automatic system must be considered. Automatic assembly systems are more demanding than manual systems, so if a part complies with the requirements to be assembled automatically, it will be possible to assemble it manually without problems.

### 6.2.2. DFA in the Conceptual Stage of the Design of the Product

The conceptual stage of the product design is the stage in which the most influential decisions are taken. The functionality of the product is defined at this stage, and it is possible to define if the product is original or variant.

Once the general functions of the product are defined, the parts that will form the product are proposed and each of these parts has to be analysed in order to determine its particular functions. The most appropriate kind of system to be used for this product can be proposed, allowing designers of the product, resources, and process to start interacting in a concurrent process. DFA considerations can be taken into account according to the kind of system selected. This stage of the product design is illustrated in Figure 6.1 using the IDEF3 Process centred description. Figure 6.2 shows the elaboration diagrams for some of the processes included in Figure 6.1. The complete set of elaboration diagrams is included in Appendix D. The elaboration diagrams provide information about the objects, facts, and constraints that the processes modelled in the IDEF3 diagrams have. The author used the Elaboration Diagrams to obtain the list of information requirements used for defining the required functionality.

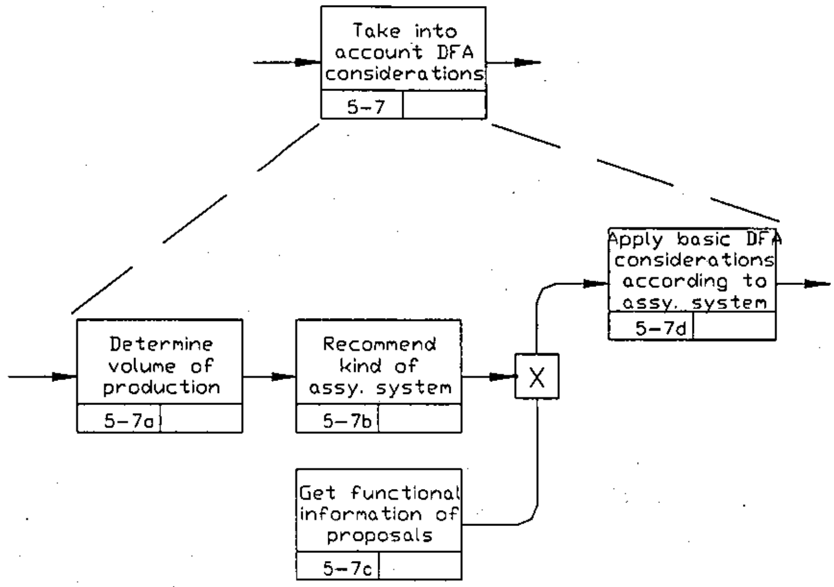



Figure 6.1 Take into account DFA considerations in the conceptual stage of the design of the product

Even though at the conceptual stage of product design there is a lack of details about the product characteristics, DFA guidelines can be applied at this stage to help the designer in the analysis of the generated concepts.

| | | | | |
|---|---|---|---|---|
| USED AT: Loughborough University | ANALYST J.M. Dorador DATE: Mar, 2001<br>PROJECT Product and Process Information Interactions in Assembly Decision Support Systems<br>NOTES 1 2 3 4 5 6 7 8 9 REV: | WORKING<br>DRAFT<br>RECOMMENDED<br>X RELEASED | REVIEWER | DATE |

| | |
|---|---|
| UOB No. 5-7a | **UOB Name:** DETERMINE VOLUME OF PRODUCTION<br>**Objects:** System User, statistics of current products, general characteristics of the product, estimates from marketing departments<br>**Facts:** The expected volume of production has to be determined in order to be able to define the kind of assembly system to be used. This estimation is done according to information from the marketing department of the company and from statistics of current products.<br>**Constraints:**<br>**Description:** The user has to introduce the estimated production volume to the system. |
| UOB No. 5-7b | **UOB Name:** RECOMMEND KIND OF ASSEMBLY SYSTEM<br>**Objects:** System User, Facilities Information, General characteristics of the product, table of recommendation for kind of system depending on product's characteristics.<br>**Facts:** The volume per hour, life of product, and number of parts determine the kind of system that can be used for assembling. The types of system are Assembly Station, Modular Assembly, Custom Assembly, Multistation Assembly, Assembly Cell, Dedicated Assembly Line, Dedicated Automatic Assembly Machine.<br>**Constraints:**<br>**Description:** The system shall evaluate the general characteristics of the product and recommend the most appropriate kind of system for assembly. |

| | | |
|---|---|---|
| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE<br>UOB Elaboration: |

Figure 6.2 Elaboration diagrams for the Units of Behaviour illustrated in Figure 6.1.

The starting point is the search, within the range of products produced by the company, of products that fulfil the same, or a similar function. This will help the designer to determine the need for the creation of a new product or to create a variant of an existing product. This applies not only to complete products, but also to subassemblies and components. Whenever possible, it is convenient to create modular components that can be used in a range of products.

If a new product or component has to be created, the designer has to identify items that require similar assembly processes, in order to highlight relevant handling and locating features. By taking into account those features in the new design, the required adaptation of available resources will be minimised.

In the creation of alternative design concepts, the designer must be aware that the use of components that are slippery, delicate, flexible, very small or very large, or that are hazardous to the handler, are more difficult to handle and insert.

It is proposed in this research to revise the generated concepts for assuring their suitability for assembly at the end of the conceptual design stage, prior to the configuration design, this activity is shown in Figure 6.3.

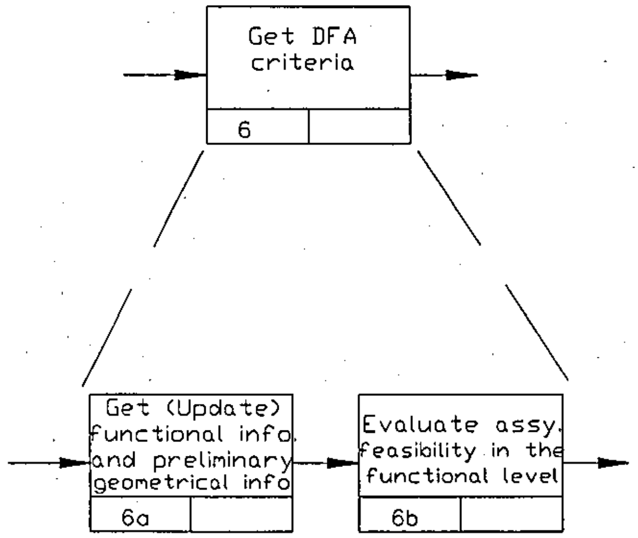



Figure 6.3 Get DFA criteria and analyse the generated concepts

At this stage of product development, the main functionality of the product has been defined. Due to the importance and effect that decisions in this early stage have on the final design and cost of the product, it is important to evaluate that functionality using assembly parameters.

Several researchers have proposed parameters for evaluating functionality at the detailed design stage of the product, however, the author considers that some of these parameters can be applied at the conceptual stage of the product design. The author has created the list below based on several researchers work, as outlined in the literature review included in Chapter 2.

- The suitability of handling and feeding of a new component must be checked after the completion of the conceptual design. If the part will be manipulated with a gripper, it must have appropriate surfaces for grasping. If the parts will be oriented using a vibratory bowl feeder, their geometry must allow that orientation.
- Avoid establishing functionality that requires tight tolerances.
- Standard and modular parts allow the reduction on manufacturing and assembly costs.
- The parts must be easily identified and differentiated from other parts of the same product or subassembly.
- Provide features to avoid jamming of parts that tend to nest, tangle or stack when stored in bulk.
- Be aware that parts that stick together or are slippery, delicate, flexible, very small or very large or that are hazardous to the handler are more difficult to handle and insert, and some times require special aids.
- Design parts that have end-to-end symmetry and rotational symmetry about the axis of insertion. If the parts they cannot be symmetric, design them to be obviously asymmetric
- The assemblies should be done following a layer sequence in a top-down straight line.
- Consider that large or heavy parts are more difficult to handle and assemble.

Once this stage of product design has finished and the embodiment stage will start, if the DFA recommendations given were applied, the solution principles selected would have embedded assembly characteristics that would make the assembly simpler.

### 6.2.3. DFA in the Embodiment Stage of the Design of the Product

It is during the embodiment design stage that Design for Assembly can make its biggest influence to obtain a good product design by providing rules, guidelines and assessment mechanisms. It is at this stage where solution principles are transformed into configurations for the product, and the main characteristics for the items that compose the product are defined. The item's characteristics will be fully specified in the Detail Design stage, but the changes in that stage are more difficult, costly and have less influence in the final design of the product.

The application of DFA rules and guidelines is done after the primary distributions have been developed and also after feasible distributions have been selected. The application of these rules is shown in the IDEF3 diagram shown in Figure 6.4.

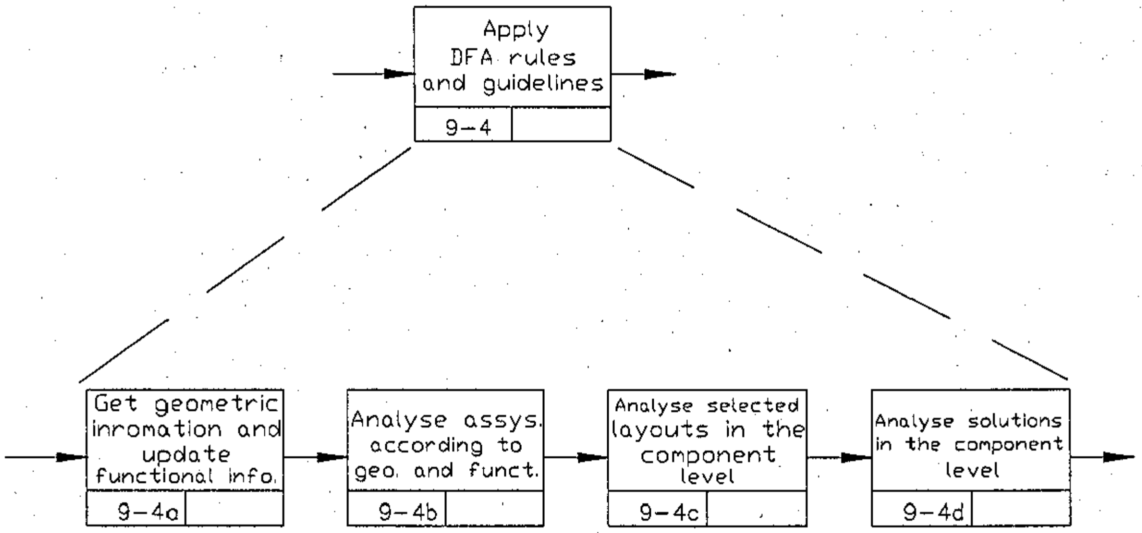



Figure 6.4 Apply DFA rules and guidelines in the embodiment design stage.

One of the most important activities in the embodiment stage of the product design is the determination of base parts for the assembly. These parts have to be stable and allow the other parts an upside-down insertion in a straight line. The correct selection of these base parts will affect the assembly of all the subsequent parts.

Using the conceptual design generated in the conceptual design stage, the embodiment design proposes the basic characteristics of the product. In order to support this activity, DFA can be applied at two levels, namely individual components and groups of parts (assemblies and subassemblies); these two levels are explained below.

#### 6.2.3.1. DFA in the Embodiment Design of Individual Components

One of the key aspects that affect assembly is the ability to grasp the component and manipulate or feed it. Although the automatic feeding of components demands more specifically designed features than the manipulation in manual assembly, in both cases the designer must consider the inclusion of characteristics that simplify those activities and by doing so the product assembly time and costs are reduced.

##### 6.2.3.1.1. Recommendations for manipulation

Manipulation is the term used when a human operator does the handling of the part. Although the human is able to perform a huge range of activities in a highly flexible way, it is necessary to provide the product with some characteristics that simplify the operations. The following list contains a collection of guidelines based on the work of several researchers, as outlined in chapter 2.

- Analyse the symmetry of the parts; if the parts are not end-to-end symmetric and rotationally symmetric propose a modification in order to achieve the symmetry. If it is not possible, propose to make the parts fully asymmetric.

- Analyse the geometry of the parts in order to detect possible problems of stacking, tangling or nesting. (Projections, holes and slots cause problems)
- Analyse geometry and material to detect parts that stick together or are slippery, delicate, fragile, flexible, hazardous to the handler, moisture-sensitive, that has magnetic properties or that are sensitive to static electricity. If the material has one or more of those characteristics, reengineer it, or place it in the assembly as late as possible.
- According to the weight and size of the parts, determine if there is suitable equipment available for the handling. Suitable features for the handling must exist in the parts, like two parallel sides that are not functional surfaces. Very small parts may require optical magnification.
- Determine the centre of gravity of the part to ensure parts stability.

###### 6.2.3.1.2. Recommendations for feeding

In addition to the recommendations given for manipulation, other parameters have to be considered to make possible the handling of the parts by a machine.

Include aids for assembly, such as:

- Smooth corners are preferred to sharp ones
- Provide chamfers or tapers
- Include features in the component so that it can be gripped and inserted without changing the robot gripper.
- The geometry of the part must allow automatic feeding and orientation.

##### 6.2.3.2. DFA in the Embodiment Design of Assemblies and Subassemblies

The DFA guidelines for components were presented in the previous section; the recommendations for assemblies and subassemblies, which are the result of two more components that have been placed together, are now shown. There are numerous factors that affect the ease of assembly, or *assemblability*, in this section the most relevant are presented.

The first recommendation given in DFA methods is to reduce the number of parts. This can be done eliminating non-essential parts by including their functionality in other parts. In order to decide if a part is essential or can be eliminated, the following 3 questions were proposed by Boothroyd (Boothroyd *et al.*, 1994):

a) Does the part move relative to all other parts already assembled during the normal operating mode of the final product?
b) Does the part must be of a different material than, or must be isolated from, all other parts assembled?
c) Does the part must be separated from all other assembled parts, otherwise the assembly of parts meeting one of the above criteria would be prevented?

Similar criteria was proposed by other authors, in the Lucas DFA method the 9 proposed questions explore more in depth the previously mentioned Boothroyd's questions.
A fourth question (Delchambre, 1996) can be included:

d) Do parts have to be separable for maintenance or replacement?

Some guidelines extracted from different methods, proposed by several researchers, outlined in section 2.4, that can be used at this stage of design are:

- The product must have a suitable base part on which to build the assembly that is located in a stable position in the horizontal plane.
- Standardise by using common parts, processes and methods across all models.
- Design so that a part is located before is released.
- Check the stability of the parts, avoiding the need of holding parts to maintain their orientation during manipulation.
- Avoid the need to reposition the partially completed assembly in the fixture.
- Avoid connections trying to locate the parts at the same point.
- Avoid adjustments.

- Use kinematics design principles avoiding overconstrained parts.
- Identify tight tolerances and surface finishes that make insertion difficult.
- Avoid adjustments after the assembly is completed.
- Check the parts for the inclusion of chamfers and guides for assembly.
- Use pyramid assembly –progressive assembly about one axis of reference- in a top-down straight movement.
- When fasteners are necessary, use the following sequence in selecting fasteners: snap fit, plastic bending, riveting, and screwing.
- Use screw points in the following order of preference: Oval point (best), Cone point, Dog point, Chamfer point, Header point, Rolled thread point (worst).
- The access and visibility for assembly operations must not be restricted.
- A visual recognition of the finished work is convenient (should look different).
- Provide barriers and sounds that indicate the termination of the work.

#### 6.2.4. DFA in the Detail Design Stage of the Design of the Product

As mentioned before, this research was focused mainly on the early stages of product development, however, the information structures researched include also support for the requirements existent in the Detail Stage of the product development cycle.

When product configurations have been selected is convenient to apply a structured method of DFA, such as Boothroyd's DFMA or Lucas DFA, for assessing the ease of assembly of the product and propose improvements and modifications at the beginning of the detail design stage, shown in Figure 6.5. This procedure will be repeated when the detailed design is finished, in order to assess the final assemblability of the product, but it is important to consider that at this final stage, the changes proposed would be very costly and have little effect on the design of the product.

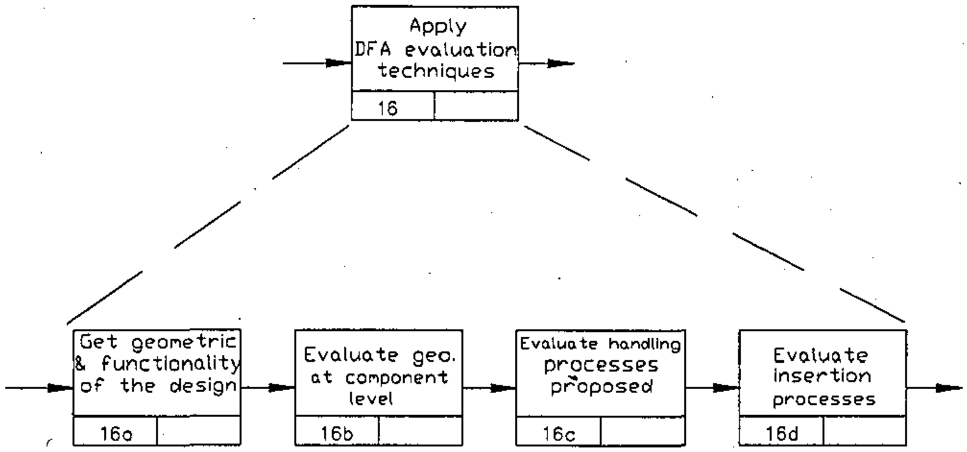



Figure 6.5 Application of DFA evaluation techniques

If the assembly recommendations have been followed in the previous stages of the product development cycle, changes due to assembly recommendations should be minor.

### 6.2.5. Design For Assembly Information Requirements

This section presents a collection of DFA information requirements through the product development cycle. These information requirements were obtained from previous sections of this chapter that evaluated other researchers' work, and from section 2.4.1. Section 2.4.1 focused on describing the functionality of DFA approaches, while previous sections of this chapter analysed the processes required for performing a DFA analysis through the concurrent development cycle by using IDEF3 diagrams and their related Elaboration Diagrams, included in Appendix D.

- Identify geometric characteristics that affect assembly:
  - Dimensions: size, volume
  - Weight
  - Symmetry (rotational and end-to-end)
  - Tolerances
  - Surface finish
  - Sharp edges
  - Chamfers
  - Natural position of the part
  - Grasping, handling and locating features
  - Mating relations (Geometric relationships between items)

- Consider functionality of the part
  - Movement relative to other parts
  - Reason for the selection of material
  - Reason for having the part as a separate part
  - Classify components in necessary and secondary
- Selection of the system
  - Annual production volume
  - Number of models in the product, generic and variant products.
  - Lot size
  - Number of parts in the assembly
  - Duration of the life of the product
  - Complexity of the product (Number of operations required)
- Identify assembly sequence (BOM)
  - Processes required
  - Assembly direction
  - Time required for adjusting the part
  - Reorientation needed during assembly
- Insertion Analysis
  - Fitting of the parts, avoid tight fitting assemblies
  - Grasping Analysis
  - Minimise manipulation and reorientation
  - Possibility of automatic orientation of the part
  - Visibility of assembly location
  - Depth of insertion
  - Barriers and sounds that indicate termination of work
  - Avoid adjustments
- Recognise:
  - Projections, holes or slots that cause tangling
  - Possible alternative ways of fitting the parts
  - Suitability for handling (appropriate surfaces for grasping)
  - Stable base part
  - Assembly in layers (top-down straight line)
  - Locate before release

- Identify possible problems for feeding the parts:
  - Problems for retrieval from bulk (jamming, nesting, stacking, tangling)
  - Difference in size of the three main dimensions (at least different in 10% to allow automatic orientation)
  - Flexible
  - Slippery
  - Hazardous parts
  - Fragile and delicate parts
  - Sticky or magnetic parts
  - Parts too small or too big.

## 6.3. Assembly Process Planning Application Functionality

This section evaluates the application of Assembly Process Planning through the process and resources development cycle in order to identify the APP information requirements. These information requirements are required for determining the attributes of the classes that compose the Information Models that support APP.

APP considers the technological requirements of the assembly to be performed in order to generate assembly tasks and sequences of the assembly operations, estimation of task time, determination of resources needed to produce the assembly and the allocation of tasks.

APP is traditionally applied when all the detailed information of the product is available, however this research analysed the role and recommendations that APP can provide in a concurrent development cycle. These recommendations influence the design and selection of the resources and processes used for assembling the product.

### 6.3.1. Application of the APP in the Design of the Process

The design of the process is the definition of the actions that will be followed for producing assemblies. In a Concurrent Engineering environment, it is necessary to have information about the processes that can be performed because these processes are developed in parallel with the design of the product and the selection of the resources.

In order to evaluate the influence of APP activities in the design of the Process, all the stages of the process development cycle were analysed. As a result, it was found that APP influences mainly three stages, shown in Figure 6.6. One is at the end of the conceptual design of the Process, the second is for the generation of assembly sequences, and the last in assessing the required time for fulfilling the operations, in order to balance the production lines.

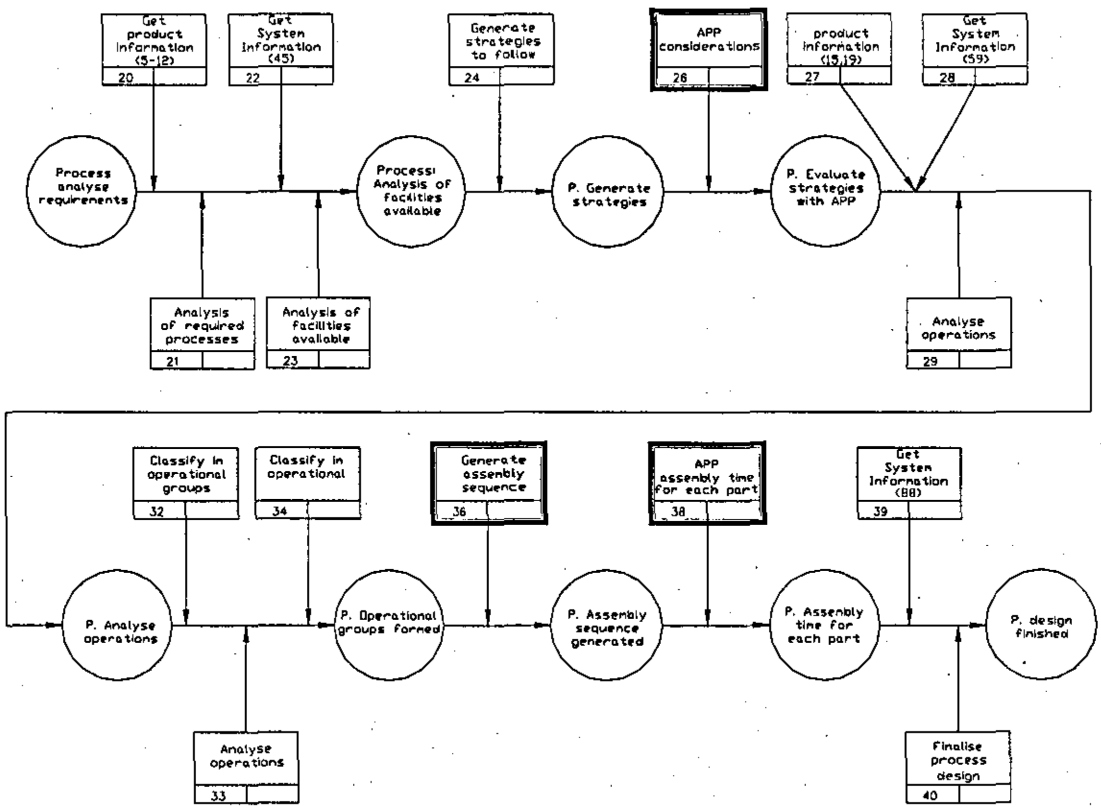



Figure 6.6 Object centred description for the process design showing the interactions with APP

In the generation of assembly strategies (activity 26), APP provides some recommendations, depending on the facilities and equipment available, and the type of system that will be used for assembly is analysed, according to the decisions taken in the product design.

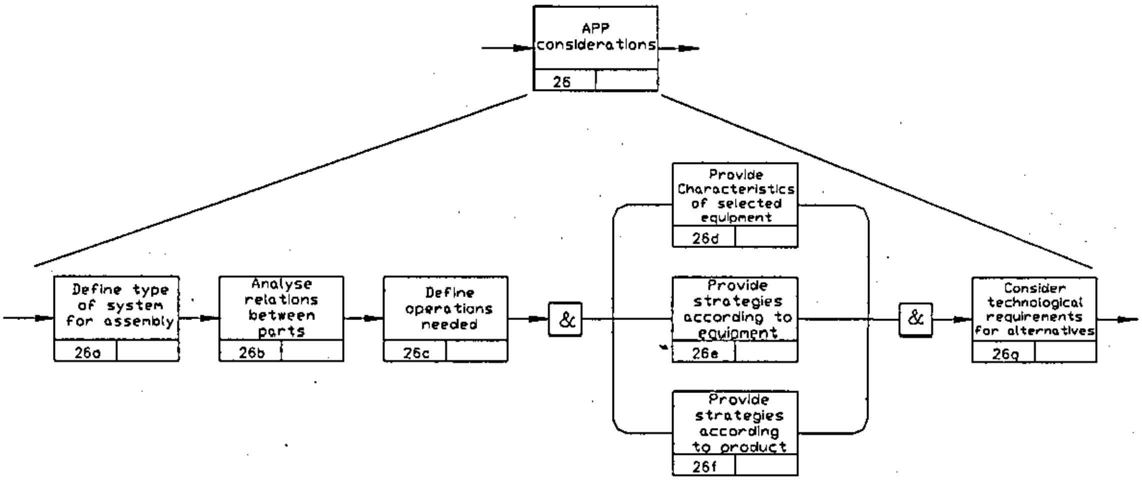



Figure 6.7 APP considerations in the conceptual design of the process

The relation between the parts is analysed, taking into account if these relations are between products that have been completely designed or if the information comes from the conceptual stage of the product design. In the latter situation, the information must be considered as temporary and very likely to change.

For the assembly plan creation an assembly tree representation or a BOM for Assembly are necessary. A base part in which other components or subassemblies will be assembled has to be defined. Using that part as starting point, the assembly tree is analysed according to the actions that have to be performed to complete the assembly of the product, considering clusters of actions and movements, times and costs for the actions.

Using information about the type of system recommended and relations between parts, the operations that are needed can be defined, either definitely or temporarily.

APP provides guidelines to select processes, according to the product characteristics, recommended equipment, and actual equipment available. The set of possible solutions is analysed according to the technical parameters and constraints of equipment and product.

When the processes have been chosen, and operational groups formed, the sequence for performing assembly activities is defined (activity 36), as shown in Figure 6.8. Several research groups have studied assembly sequence generation.

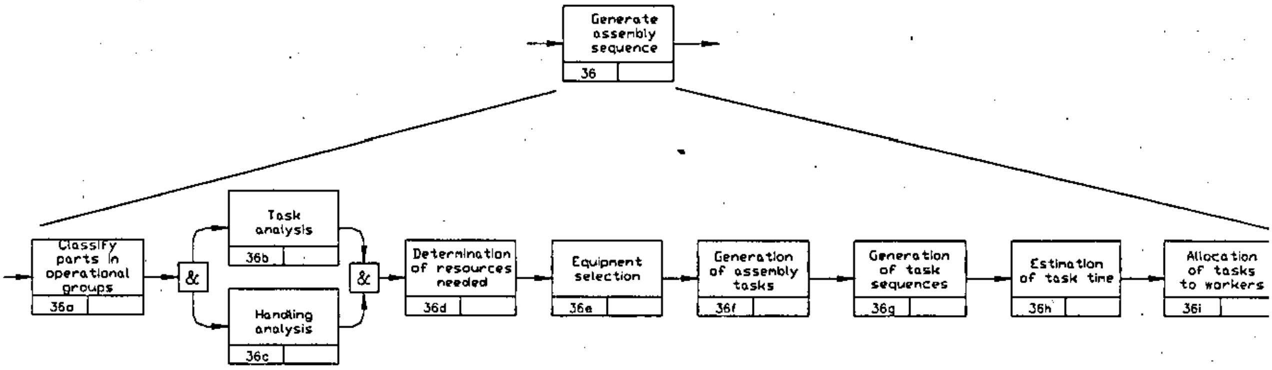



Figure 6.8 APP generate assembly sequence in the process design

Assembly sequence generation is a key activity in APP due to the benefits that can be obtained in the reduction of time and resources necessary for performing assembly when the optimum sequence is chosen. The generation of Assembly Sequences is out of the scope of this work, however, the main activities have been identified to ensure that the information models will be able to provide support to these activities.

In the design of the process, activities are classified in operational groups; this information is the basis for analysing tasks and handling operations that will be performed. Having this information, the necessary resources can be defined in the equipment selection, interacting with the system design process in order to either choose or recommend new equipment.

Once the best sequence has been chosen, the assembly time for each part is calculated and a precedence diagram for the assembly sequence is produced. These times are necessary for balancing the lines and assessing the indirect costs of the assembly, and tasks can be allocated to workers and stations. The sequence of activities is shown in Figure 6.9. The definitive allocation of tasks to the workstations and systems is done and and process sheets are issued.

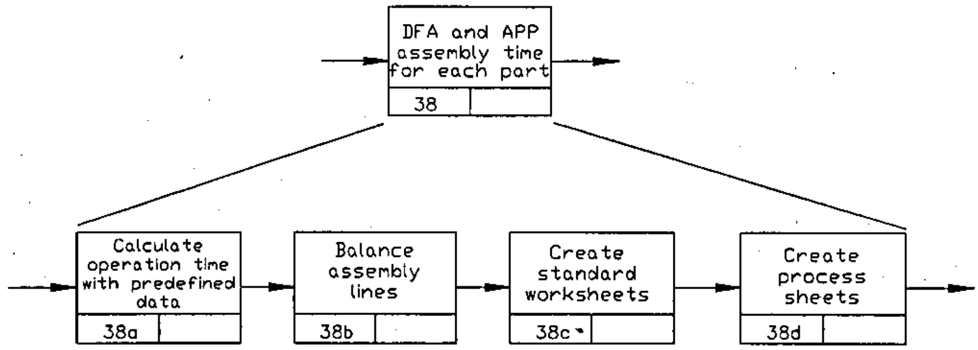
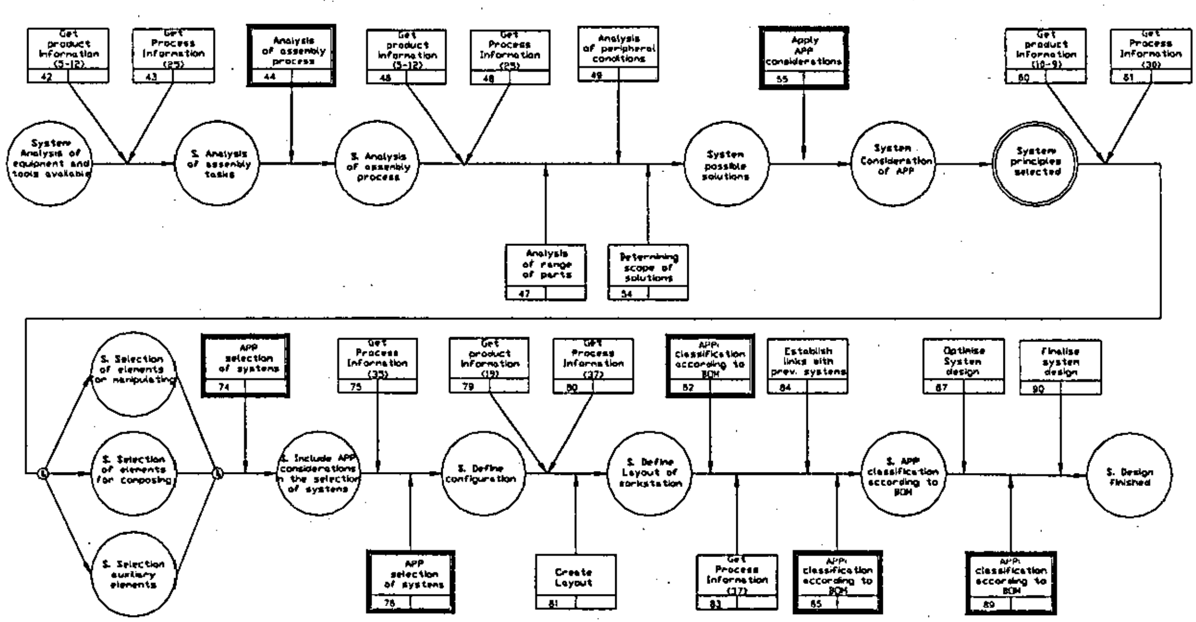



Figure 6.9 DFA and APP assembly time definition for each process

### 6.3.2. Application of the APP in the Design and Selection of the Resources

In the design and selection of the assembly Resources, APP has several interactions with the concurrent design process, as shown in Figure 6.10. At the beginning of the design cycle, APP analyse the required processes. When some solutions have been generated, APP considerations must be taken into account in order to select the best option. APP has a significant role in the selection of systems and equipment. Once the configuration and layout have been defined, APP helps to assess them. APP is also used for establishing relations between product and systems, according to the Bill of Materials.



#### 6.3.2.1. Application of APP in the Conceptual Design Stage of Design and Selection of Resources

In order to start the design of resources, it is necessary to obtain information about the product and the assembly processes required for assembling it, as depicted in Figure 6.11.

According to the product and process design information, the required equipment for performing the operations is proposed. The existing equipment is analysed in order to decide if the assembly operations can be performed with the existing resources.

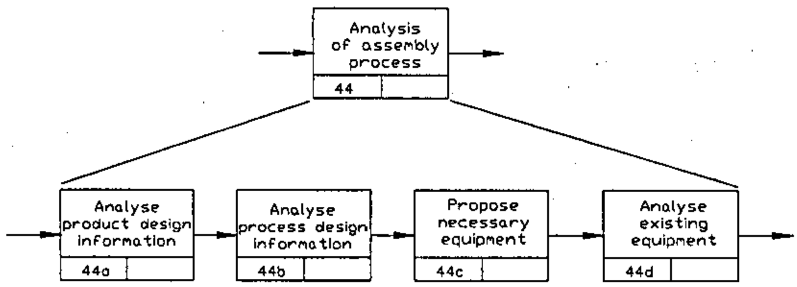



Figure 6.11 APP analysis of assembly process in the conceptual design of resources

When the possible solutions for the system have been produced, APP considerations help to select the principles that will be followed, as shown in Figure 6.12. The scope of solutions proposed is evaluated against economic, process and systems constraints.

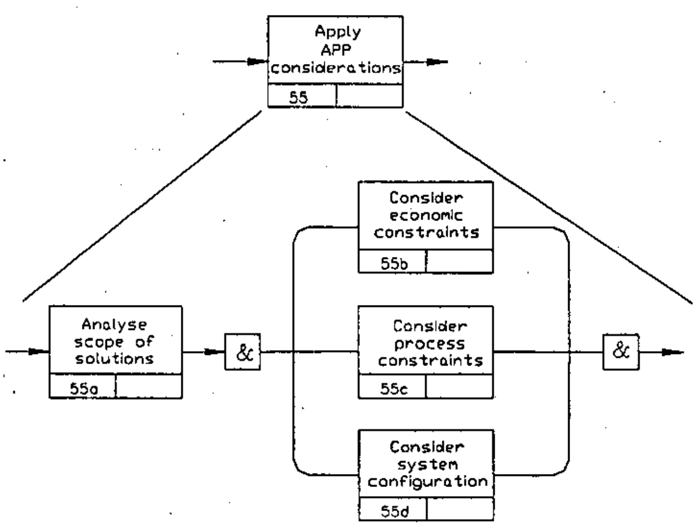



Figure 6.12 Apply APP considerations in the conceptual design of the resources

### 6.3.2.2. Application of APP in the Embodiment Design Stage of Design and Selection of Resources

In the embodiment design stage of the design and selection of resources, APP supports the selection of assembly systems (activities 64, 69, 74 and 78 in Figure 6.13). APP is the link between task requirements and the Manufacturing Model that contains information about the resources available. During this stage, it will be decided if is necessary to acquire or design new equipment, tools or jigs for performing the assembly.

For the selection of manipulating, composing, and auxiliary elements the APP evaluates the task and process requirements from the product design information, the characteristics of the equipment available and the possibility for interfacing the equipment.

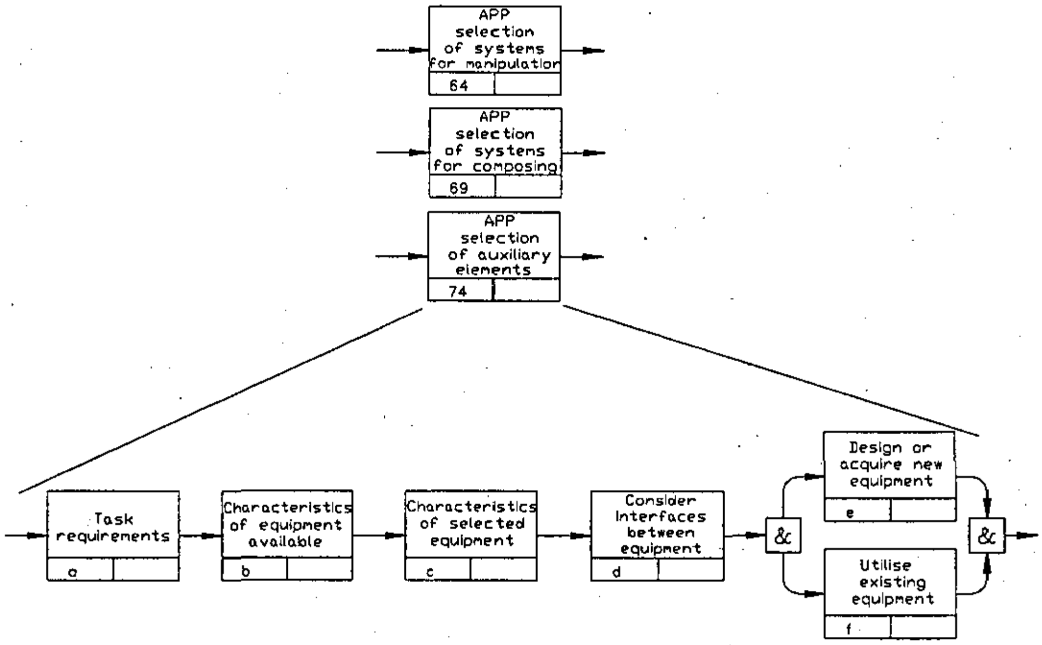



Figure 6.13 APP support for the selection of systems in the embodiment design of resources

Once the equipment is selected, it can be grouped in cells and stations, helping the process of creating and analysing layouts. Process information and time parameters have to be considered in order to create these groups and balance lines. These activities are depicted in Figure 6.14.

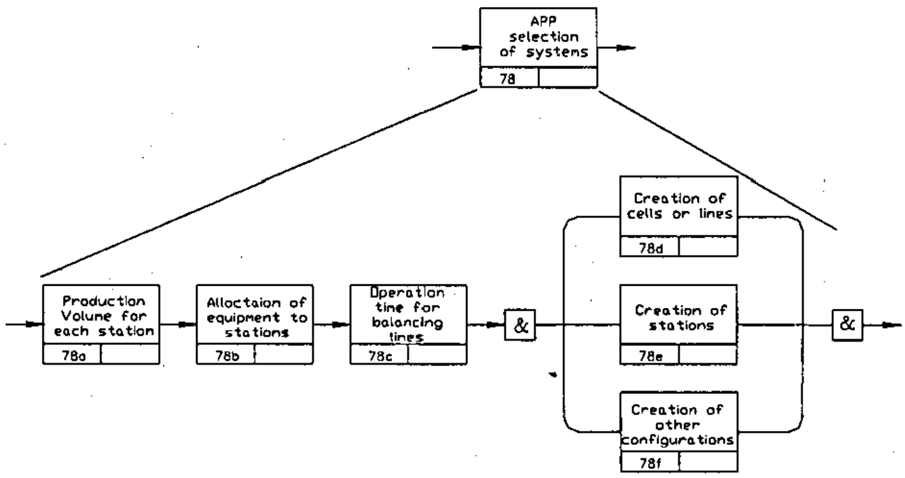



Figure 6.14 APP selection of systems for the configuration definition of resources

### 6.3.2.3. Application of APP in the Detail Design Stage of Design and Selection of Resources

In the final stages of the design and selection of resources, APP helps to classify the resources to be used according to the BOM (activities 82, 85 and 89 in Figure 6.15), providing the necessary information to the shop-floor controller, and by creating standard worksheets for performing assembly activities.

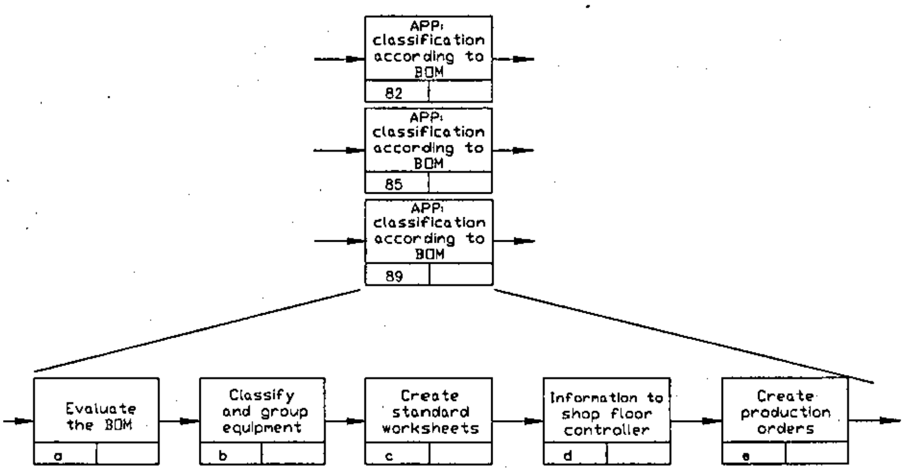



Figure 6.15 APP classification according to Bill of Materials

During the actual assembly process, APP helps the shop floor control by changing the pitch time and the number of assembly process according to the change of production volume.

### 6.3.3. Assembly Process Planing Information Requirements

As a result of the evaluation of the Assembly Process Planning activities through the Process and Resources development cycle, it can be said that it requires similar information as DFA about the components involved in an assembly, their relationships and interdependencies. DFA information requirements are listed on section 6.2.5. However, the results obtained from the application of DFA and APP are completely different. DFA relates the information models to the difficulty associated with assembly and computes the assemblability of the design. APP results are mainly about the detailed assembly plans that can be used for production.

Besides the information requirements listed for DFA in section 6.2.5, Assembly Process Planning has the following information requirements.

- Information of the resources available
  - Transporting resources
  - Assembly equipment
  - Storage equipment
  - Capabilities
  - Limitations
  - Location of the resources
- Information about processes
  - Operations required for performing the assembly
  - Handling processes available and their capabilities
  - Fitting processes available and their capabilities
- Information about the strategies
  - Which processes can be performed with the resources available
  - Methods of transporting
  - Methods of insertion
  - Methods of storage
- Information of the product
  - Assembly sequence
  - Definition of the base part
  - Processes necessary for assembling the part
  - Assembly process plans for other parts
  - Precedence relations between parts

The information structures of the Product and the Manufacturing Models are detailed in chapter 7. The application software designed for testing the use DFA and APP with Information Models is detailed in chapter 8.

## 6.4. Summary

This chapter evaluated the influence that DFA and APP have throughout the concurrent development process of the design of the product, and the processes and resources for assembly. Emphasis was set in the early stages of the development process, in which DFA and APP are normally not intensively applied in spite of being the most influential stages.

The approach followed in this research has the potential of improve the ways in which DFA and APP are applied by using Information Models to support them. It has been found that the use of Information Models allows the access to information that is being generated concurrently and by doing so, the recommendations given by DFA and APP can interact in all the stages of the development of the product, the process and the resources.

The information requirements found for DFA (section 6.2.5) and APP (section 6.3.3) were identified, and used to define the attributes for the classes that constitute the structures for the information models, detailed in chapter 7.

The IDEF3 process centred description for the design of the product, processes, and resources allowed the description of the functionality of the DFA and APP. This information was used as a basis for creating the requirements trace matrix for the development of the testing software, which will be explained in chapter 8.

# 7. DEFINITION OF THE STRUCTURES FOR THE PRODUCT AND MANUFACTURING MODELS

## 7.1. Introduction

The aim of the research that is reported in this thesis was to make a contribution in the concurrent engineering area by providing structures for Information Models to make them able to support assembly-related activities.

This chapter defines the information structures for the Product and Manufacturing Models. The general structures of these information models were given in chapter 4, and the detailed structures of the models are explained in the first sections of this chapter. These structures are the result of the application of the methodologies described in chapter 5 and the evaluation of the information requirements imposed by DFA and APP, described in chapter 6.

The detailed structures of the Product and Manufacturing Models are explained in sections 7.2 and 7.3. These structures support interactions between the Product and Manufacturing Models, as explained in section 7.4, and are a source and repository of information for Design For Assembly and Assembly Process Planning activities.

Though the structures for the Information Models defined are generic, the author recognises that these models will require modification in particular industrial applications. The scope of the model is targeted at electromechanical products, however the basic structures can be extended to capture other kind of products.

## 7.2. Information Structure for the Product Model

The Product Model concept and its background were analysed in section 2.3.2 of this thesis, where it was defined as "a computer readable representation of all product related data" (Young and Bell, 1995). This representation of product data contains detailed information about the product, so it can support software applications that interact through the concurrent development cycle by providing access and being a repository of such information.

The general structure for the Product Model was given in section 4.4. This section analyses the detailed structure defined for the Product Model, which is depicted in Figure 7.1. The structure proposed for the Product Model is an important result of this research because it has the ability to support assembly-related activities, according to the information requirements established in chapter 6.

The author's structure for the Product Model extends existing work by including in an integrated structure the Characteristics of the Product, the Product Range and the Views from which the information can be represented. These aspects of the Product Model are represented in Figure 7.1, the shaded classes in the figure are those taken from the structures defined in the MOSES project.

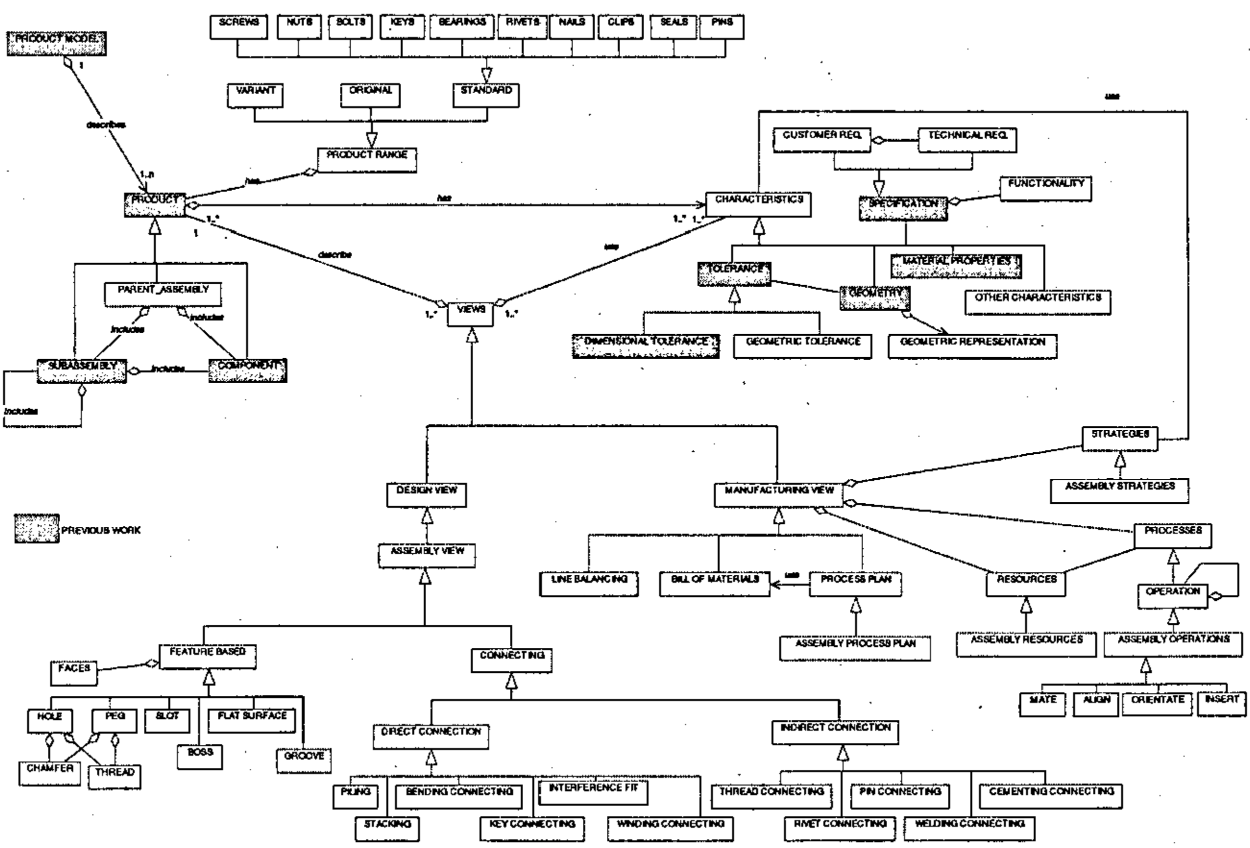



Figure 7.1 Extended general structure for the Product Model

Figure 7.1 shows the extended general structure for the Product Model obtained in this research. Figure 7.2 shows a section of Figure 7.1, in it can be seen that the Product Model *describes* the product, the original, standard, and variant product ranges that are child classes of the parent product range inherit the structure of the abstract Product class. The inheritance relationship in an object-oriented model is a key factor in the representation of the structure of the Product Model. Parent-child relationships between classes are shown using the *is_a* relation in the class diagram such that the original, standard and variant ranges *is_a* product range.

A Variant Product is defined as a product that is part of a family of products, sharing characteristics of the family of products. The Standard product is referred to those parts, subassemblies of parent assemblies that are commercially available, such as screws, nuts, bolts, and bearings. The Original Range includes products designed according to customer specifications.

Figure 7.2 shows the Ranges and the child classes of Product, these classes are Parent Assembly, which refers to a complete product, this class includes Subassemblies that can be composed of other Subassemblies and Components. The Components are considered to be the basic unit that forms an assembly.

According to the inheritance relationships shown in Figure 7.2, the Parent Assembly, Subassembly and Components can be either Variant, Original or Standard Products, all of them have Characteristics and can be described using Views.

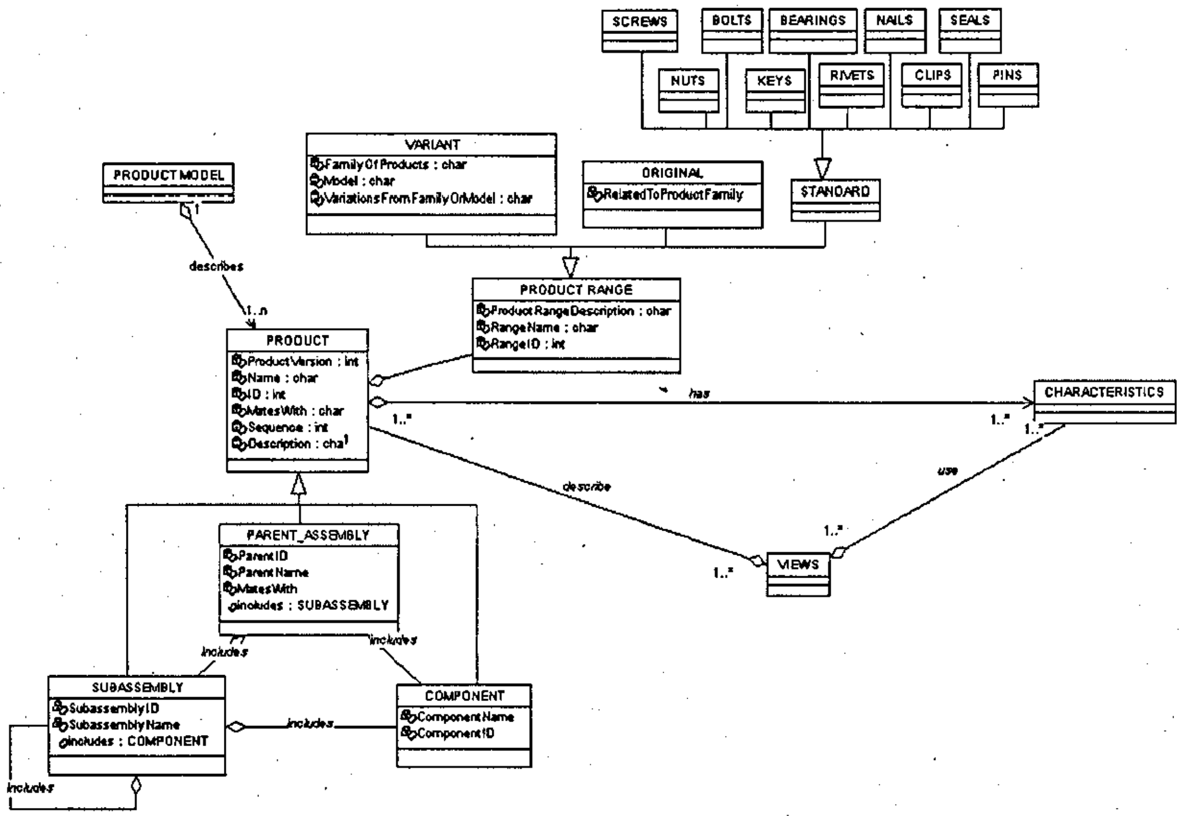



Figure 7.2 UML diagram of the Product Model showing the Ranges and children classes of Product

### 7.2.1. Characteristics Class of the Product Model

Figure 7.3 shows how the products are explained in terms of their characteristics: specification, geometry, tolerances, material properties, and other characteristics; i.e. product *has* characteristics.

The "Geometry" characteristic will be used here as an example to explain the application of the abstract Characteristics class. The Geometry Characteristic *is_a* Characteristic. The Geometry class inherits all the attributes and methods of the Characteristics class.

Therefore, the product class has access to the attributes captured by the instances of the Characteristics class, e.g. so all the instances of product will have the attributes contained in the geometry class. The parent, subassembly and component classes shown in Figure 7.2 are child classes of product, i.e. Parent, Sub-assembly and component *is_a* Product. As a consequence, they also have the capability of accessing the information captured in any instances of the Characteristics class, as shown in Figure 7.3.

The Characteristics specification class refers to the set of customer and technical requirements that must be satisfied, so it defines the functionality that must be fulfilled by the Component, Subassembly or Parent item.

All the characteristics were grouped under a virtual Characteristics class, allowing with this a more organised way of structuring the classes and also the possibility of using the Characteristics information from multiple View Points, this are explained in section 7.2.3.

### 7.2.2. Mating Conditions Class of the Product Model

An assembly can be defined with the geometry of its components and their relationships. These relationships in the assembly are called mating conditions. If only the components were considered, that information would be useful to some applications, such as the creation of a bill of materials, but not for other applications which require information about the relationships between components, such as DFA and tolerance analysis. The Mating Conditions were added as a subclass of the *Other Characteristics* class. In Figure 7.3 a representation of the mating conditions characteristics is shown.

The geometric mating conditions can be defined by the kind of relation between the parts, including fits, contact, interference and non-contact constraints. The relative position can be parallel, perpendicular, angular, collinear, offset or planar.

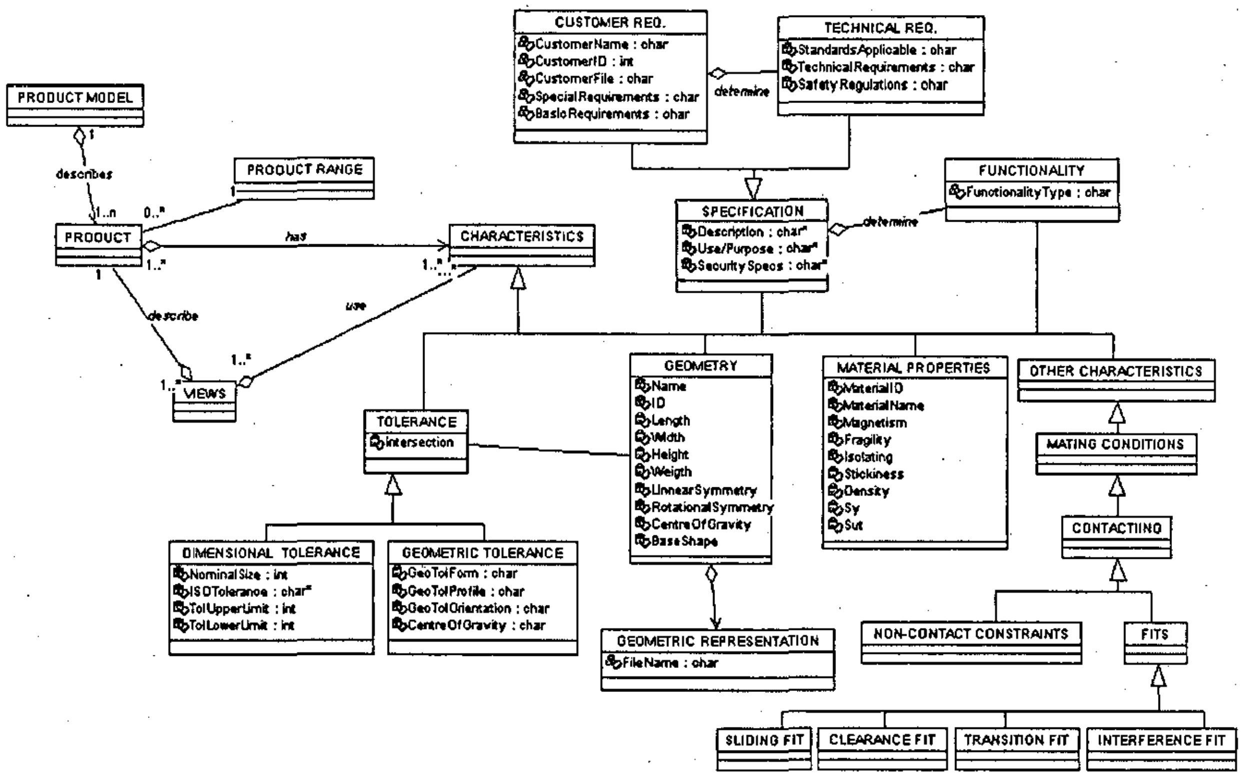



Figure 7.3 UML diagram of the Product Model showing the Characteristics

### 7.2.3. Multiple Views in the Product Model

In order to provide appropriate support to all the software applications that use the Product Model as a source and repository of product information, and allow their interactions, it is convenient to be able to access the information from different points of view. This leads to a multiple view Product Model. The Views class shown in Figure 7.4 represents this capability. The relationships that the Views class has with the Product and Characteristics classes can be read as "a View describes a product using its characteristics". Due to the inheritance existent in the model's structure, the View can describe different levels of the product (Parent Assembly, Subassembly, and Component) and different ranges (Variant, Original, and Standard).

The Design View and the Manufacturing View are among the different views that can exist. In Figure 7.4 the Design View is shown with its subclasses Design For Manufacture (DFM), Assembly, and other views. The Manufacturing View is decomposed in Line Balancing, Bill of Materials and Process Plan views. The assembly-related subclasses of the Design and Manufacturing Views are depicted in Figure 7.5 and Figure 7.6.

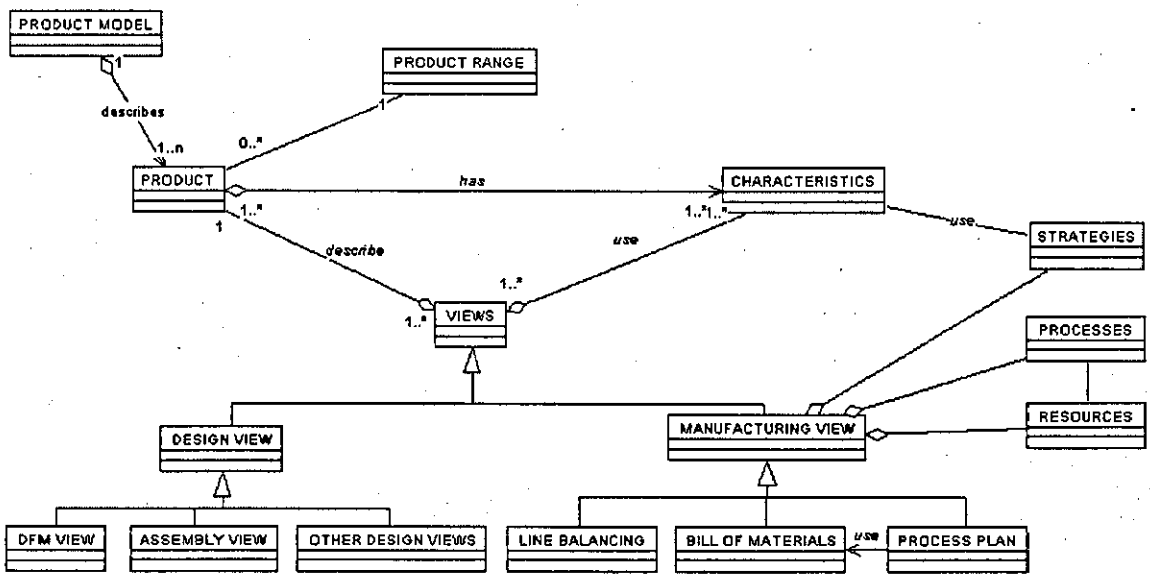



Figure 7.4 UML diagram of the Product Model showing the Views

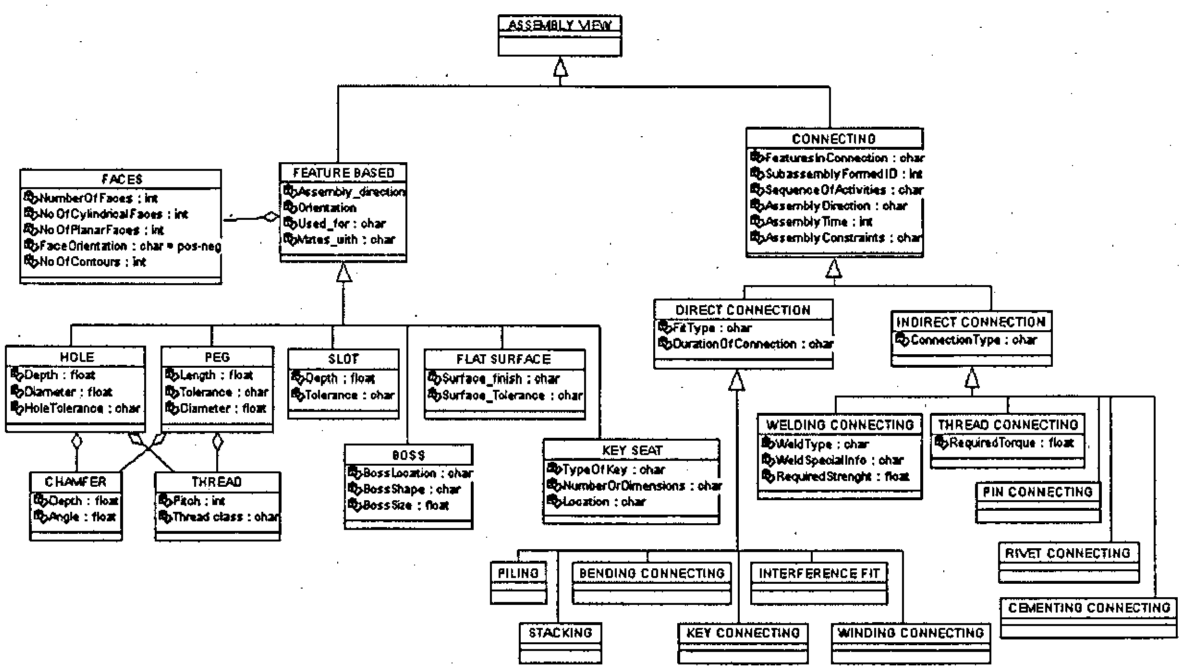



Figure 7.5 UML diagram showing the Assembly View (Design)

The Design Assembly View, shown in Figure 7.5, shows the Feature-based and the Connecting classes. In the Connecting class the relations between the parts are evaluated, according to the kind of connection between the parts, this can be direct or indirect. In the direct type, connection agents are not necessary, some examples are piling, stacking, bending, key connecting, interference fit, etc. Examples of the indirect connection, in which external agents are necessary, are thread connecting, pin connecting, rivet connecting, cementing, and welding.

In Figure 7.6, the Manufacturing View of the Product Model is shown. This view holds manufacturing information particular to a product. The subclasses shown are the Bill of Materials and the Assembly Process Plan. There is a *has* relationship with the class Manufacturing Model Information, this relationship is the key for the interactions between the Product Model and the Manufacturing Model, which are explained in section 7.4.3.

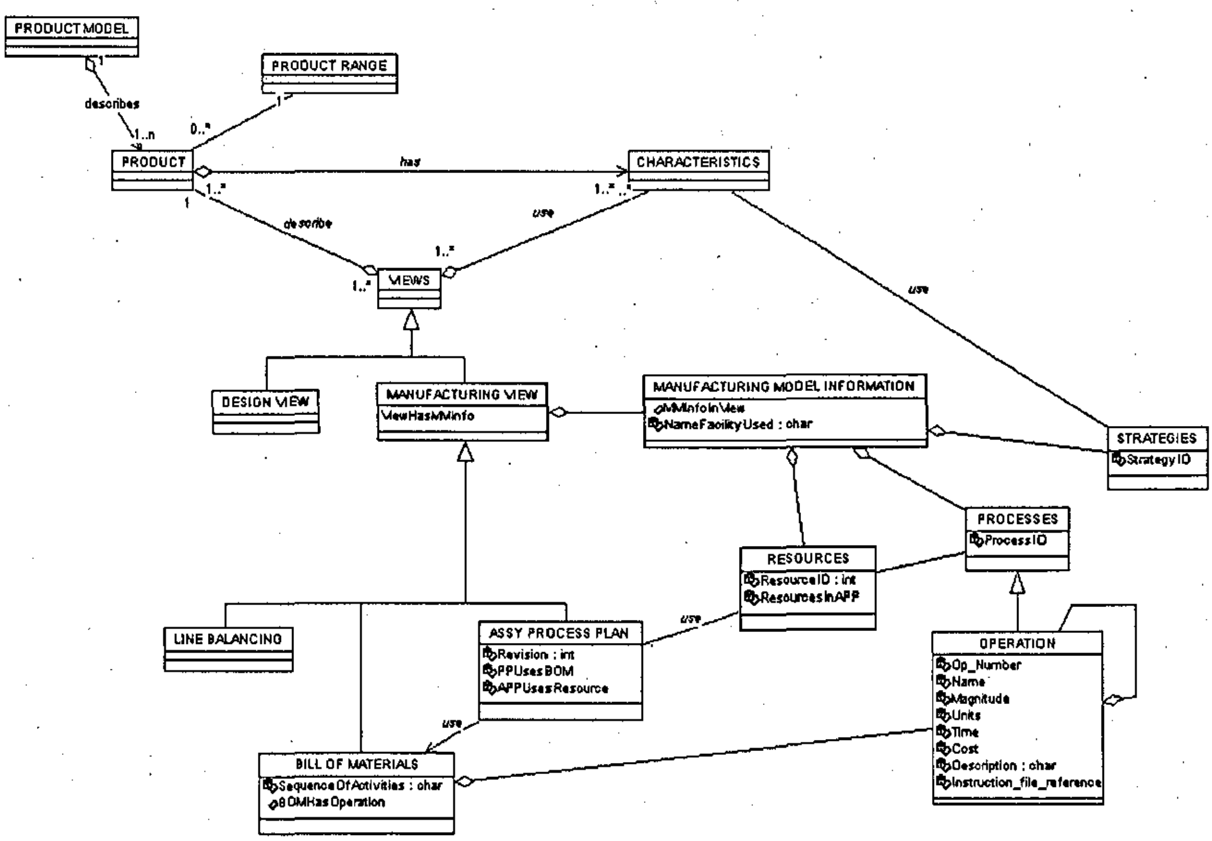



Figure 7.6 UML diagram showing the Manufacturing View of the Product Model

## 7.3. Information Structure for The Manufacturing Model

As stated on section 2.3.3, a Manufacturing Model contains information that describes the manufacturing situation of a particular enterprise to provide designers and manufacturing engineers with a source and repository of information on which to base their decisions.

The general structure for the Manufacturing Model was given in section 4.4.2, and was depicted in Figure 4.4. The extended general structure showing assembly processes, resources, and strategies is depicted in Figure 7.7, which is explained below. This structure is able to support assembly-related activities is an important contribution that extends work previously done in Manufacturing Models, explained in section 2.3.3. The starting point for this research, adopted from the MOSES project is identified in Figure 7.7 with the shaded classes.

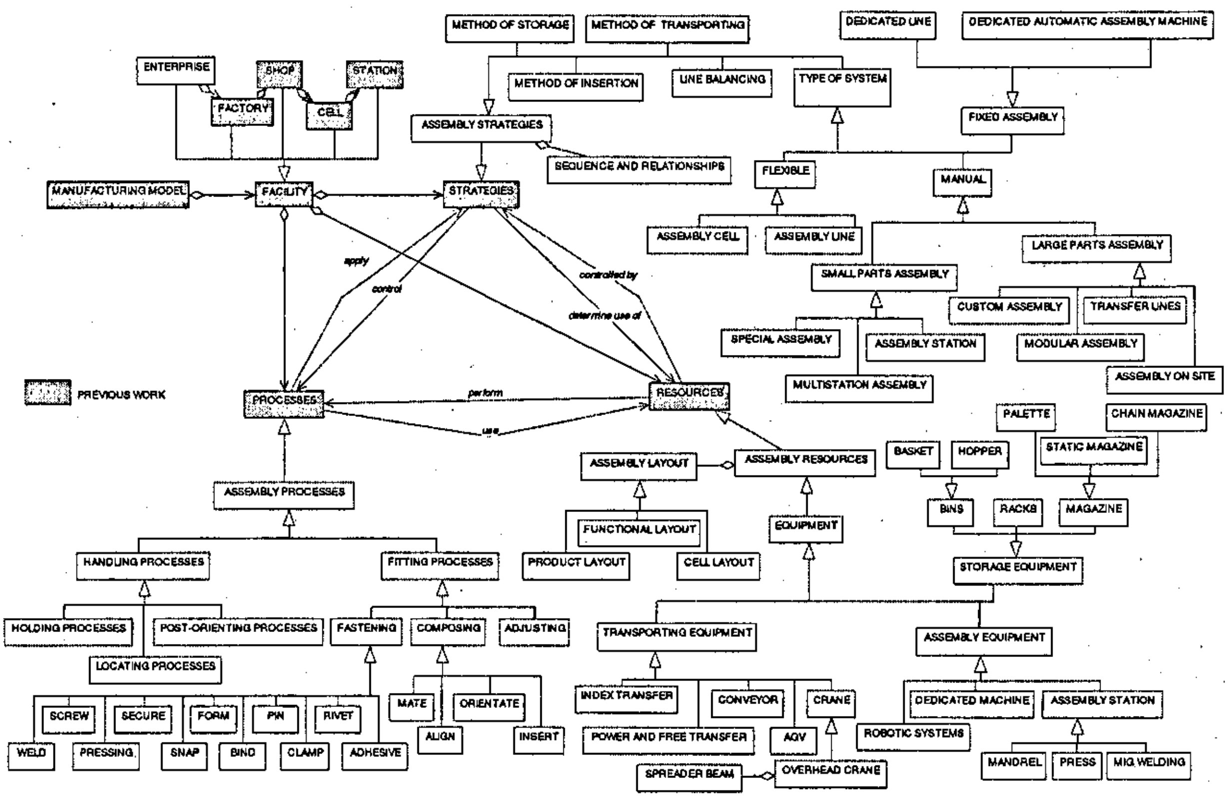



Figure 7.7 Extended general structure for the Manufacturing Model

The Manufacturing Model describes the structure of the manufacturing facilities. A facility may comprise an enterprise, a factory, a shop, a cell or an individual station. The facilities are explained in terms of their resources, processes, and strategies. These concepts are illustrated in Figure 7.8 and explained below.

The Facilities described by the Manufacturing Model can be divided in hierarchical levels, namely Enterprise, Factory, Shop, Cell and Station, they are represented by classes in the UML diagram shown in Figure 7.8. The relationship between these classes is an aggregation relationship, which shows a 'whole and part relationship' between two classes, so an Enterprise *has* Factories, which in turn *has* Shops.

The Resources are the available means within a facility. These Resources include the equipment and tools necessary to enable the product realisation. Processes are the sequences of actions necessary to produce a part or a product. Together, the Resources and Processes provide a consistent representation of the capabilities that a Facility has, and therefore, what the Facility is able to produce.

Strategies are the way in which decisions are made on the use and organization of Resources and Processes, in order to achieve the Facility's manufacturing objectives. Depending on the level in which the strategies are applied, they can be related from strategic decisions to operational rules.

The bidirectional relations among Resources, Processes and Strategies are shown in Figure 7.8. Strategies control Processes, and Processes are applied using Strategies. Resources are controlled by Strategies, and Strategies determine the use of Resources. Processes make use of the Resources, and Resources perform the Processes.

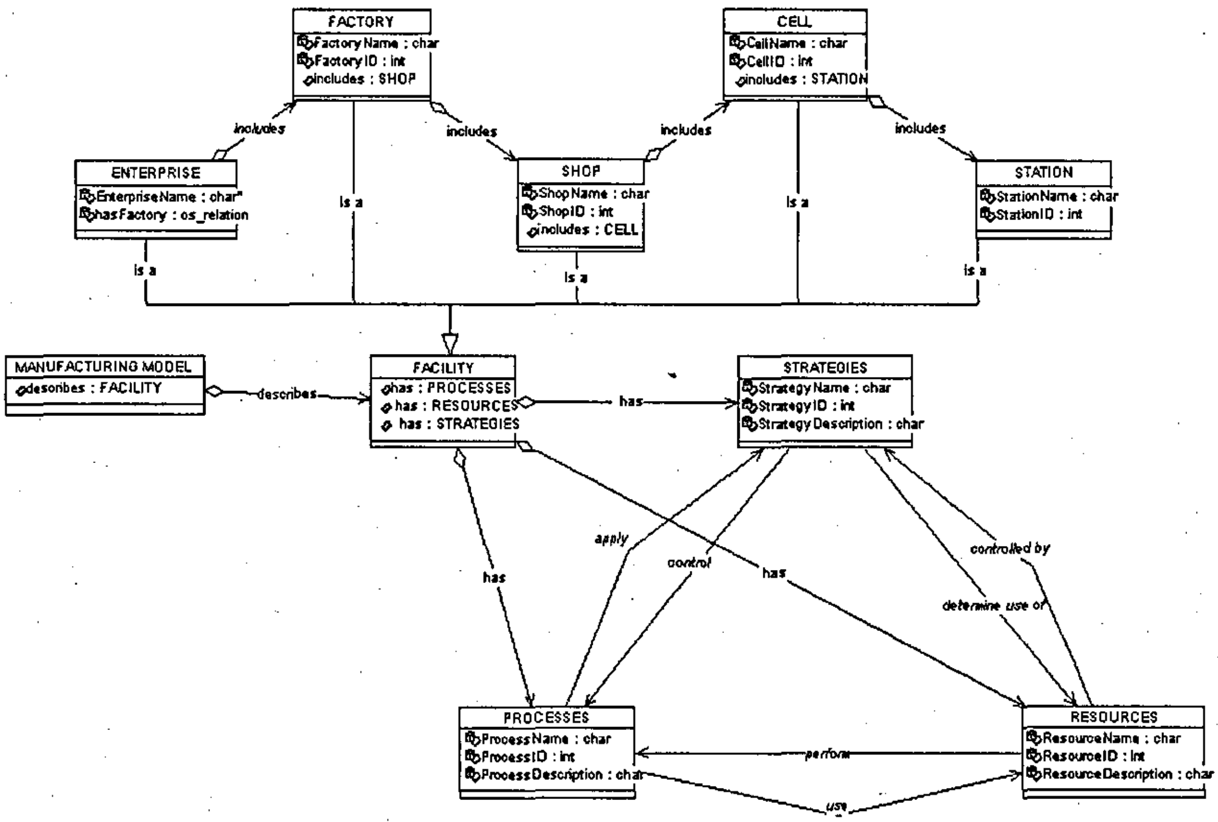



Figure 7.8 Basic structure of the Manufacturing Model

The Resources class can be classified in different kind of resources, such as Material Resources, Machining Resources, Tool Resources and Assembly Resources. Figure 7.9 shows the subclasses proposed for the Assembly Resources class, which can be divided into Transporting Equipment, Assembly Equipment, and Storage Equipment classes. The classes below that level, although general, were chosen specifically to model assembly resources used in the collaborating company, from which the real data for testing the system was obtained. The inclusion of Mandrel and Press classes as children of the class Assembly Stations are examples of classes included for modelling specifically the electrical large machine assembly. These Assembly Resources classes can be extended in order to include other assembly resources.

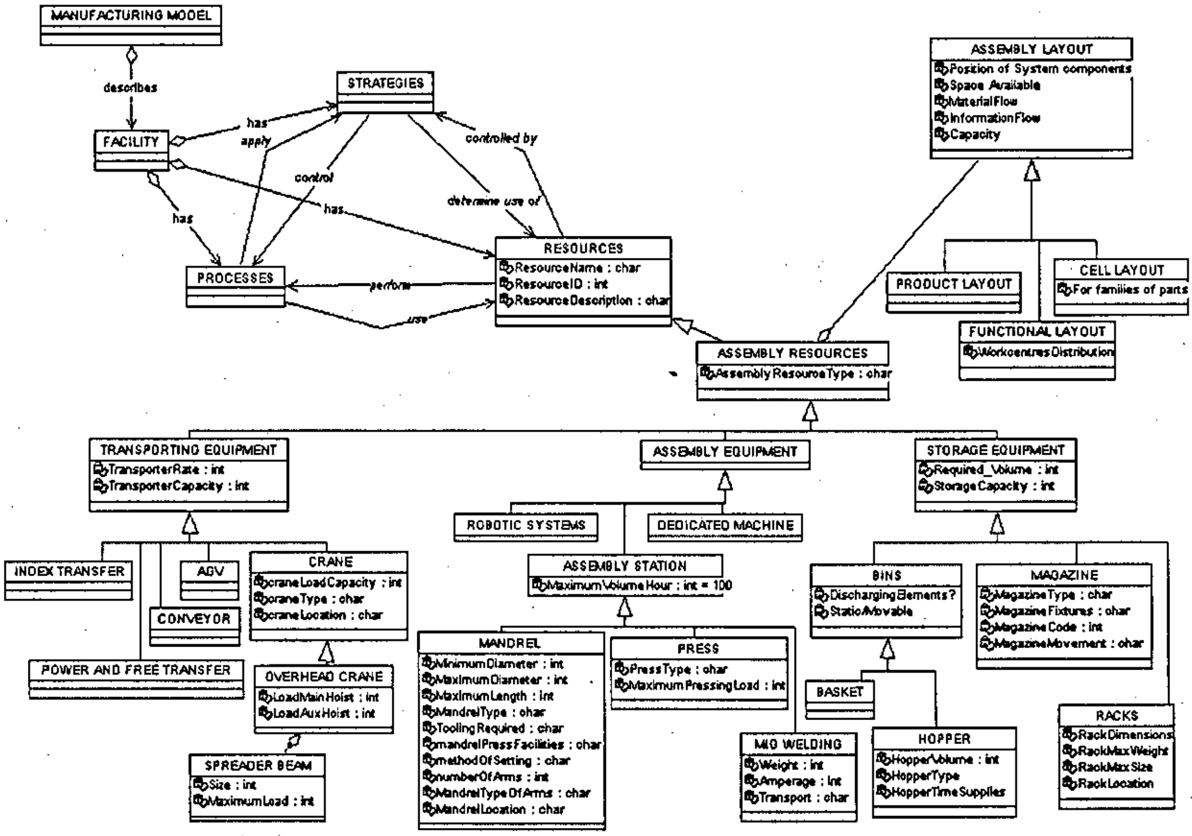



Figure 7.9 UML Diagram showing the Manufacturing Model Resources classes

A product is the result of a series of assembly processes that join subassemblies and components into a parent assembly. Subassemblies are intermediate assembly operations where a component or subassembly is assembled with another component or subassembly. Parent assembly refers to the finished product.

Figure 7.10 shows the Assembly Processes class as a child of Processes class. The general actions necessary for performing an assembly can be classified as handling and fitting, so the Handling Processes and the Fitting Processes classes are included.

Handling Processes refer to the selection and preparation of components for being inserted, so the attributes for the Handling Processes class are, besides the name, description and identification number for the operation, the sequence of activities, handling distance, initial and final position and orientation. The children classes of Handling Processes included in Figure 7.10 are Holding, Locating and Post-orienting processes. The attributes of Holding Processes are the force and the time that the component has to be held. Locating Processes need a reference for locating, so it was included as a class' attribute. The Other Processes class can include other processes required for the locating of the component, such as checking and aligning processes.

The Fitting Processes class includes the children classes Composing, Adjusting, and Fastening. Composing refers to the assembly operations in which a connection between the components is made, the possible Composing process include Mate, Align, Orientate, and Insert. The class Composing has the attribute Sequence For Composing, which holds information about the order in which that process take place in the complete set of assembly operations.

Figure 7.10 shows the children classes of the Fastening class. Fastening includes those operations that form a permanent connection between the parts.

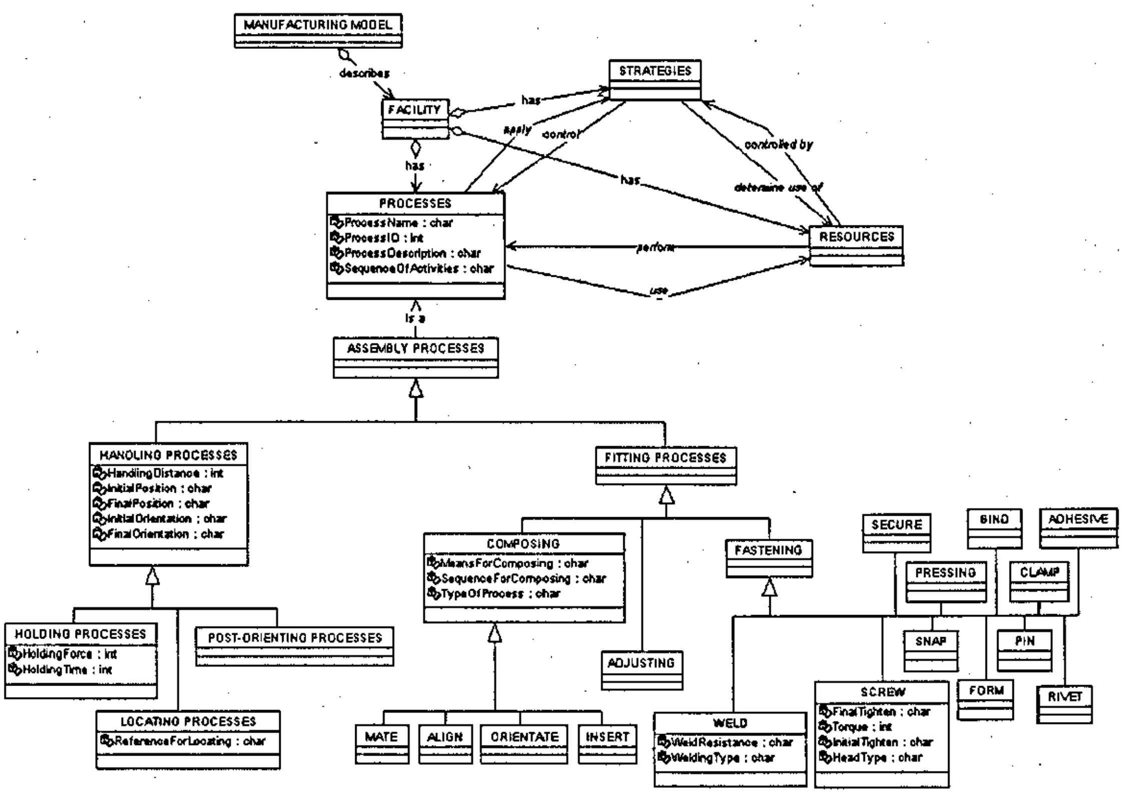



Figure 7.10 UML Diagram showing the Manufacturing Model Assembly Processes classes

Assembly Strategies are the way in which decisions are made regarding Assembly Processes and Assembly Resources. Figure 7.11 shows the children classes of the Assembly Strategies class. The classes Method of Insertion, Method of Transporting and Method of Storage can provide information for an adequate selection of the necessary Processes and Resources, so these can be considered strategies acting at a Cell or Station level. The classes Type of System and Line Balancing give recommendations at a higher level in the Facility structure, in this case, these strategies influence the Shop or Factory levels.

Figure 7.11 shows the aggregation relationship between Assembly Strategies and Sequence and Relationships. The Sequence and Relationships class holds the information about the relation existent between the parts and the order in which they are assembled.

The class Type of System is shown in Figure 7.12. This class provides the criteria for selecting the type of assembly system according to parameters explained in section 6.2.1. Some of these parameters are the expected life of the product, the volume that will be produced, and its estimated weight and size.

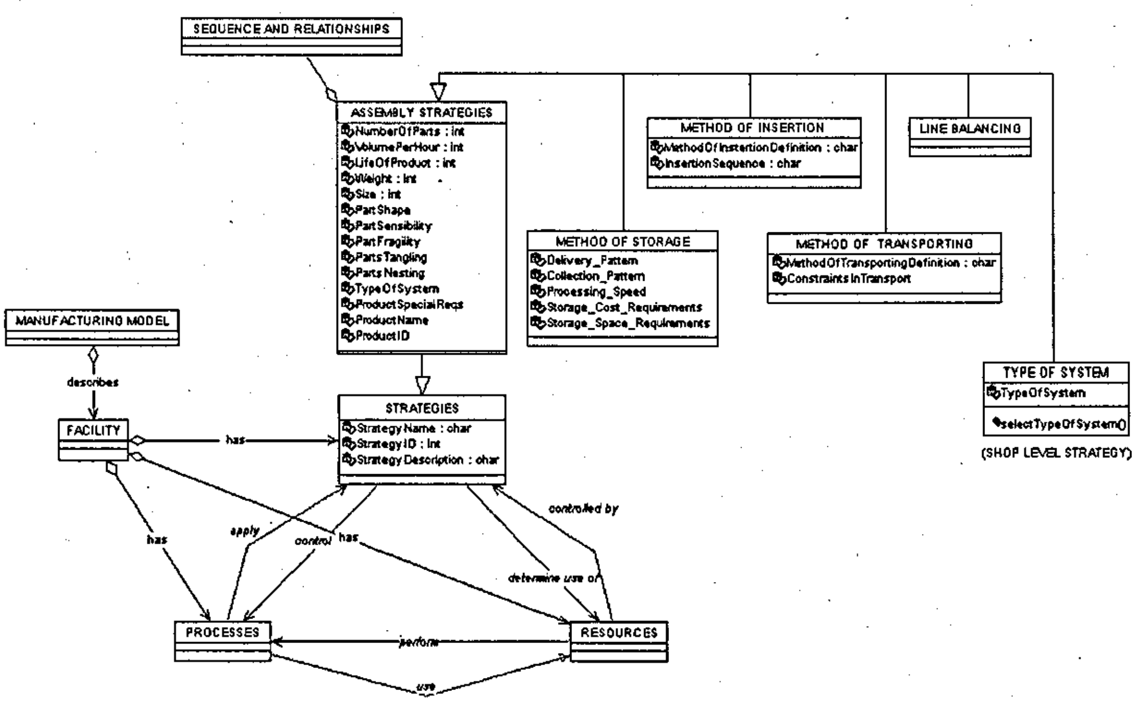



Figure 7.11 UML Diagram showing the Manufacturing Model Assembly Strategy classes

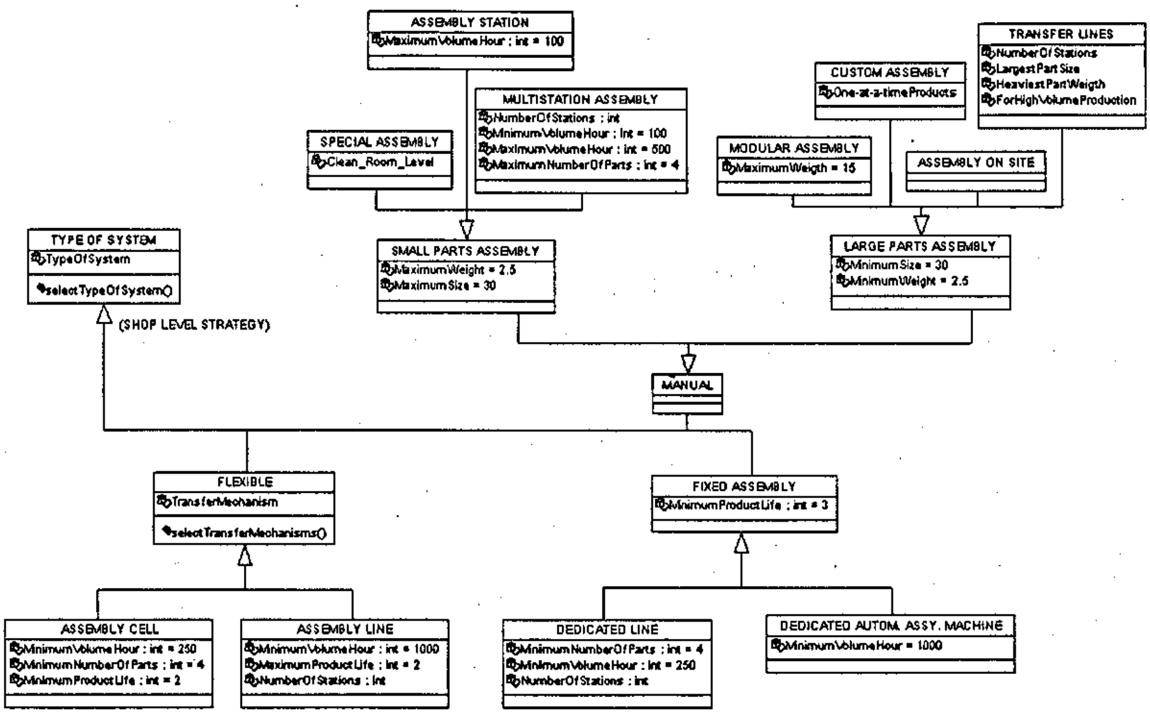



Figure 7.12 UML Class diagram for the Type Of System Strategy

## 7.4. Interrelations between the Product Model and the Manufacturing Model

In order to be able to support the Concurrent design of the product, process and resources this research followed an integrated approach, in it there is a dependency between the characteristics of the Product with the Facilities that will be used for producing it. Decisions taken during the design of the product, process, and in the selection of resources influence each other, and is likely that if there is a change in one or more of the variables, the other variables have to change.

The integral approach proposed by Rampersad, explained in section 2.4.5.7, considers that the assembly process is a function of the properties of both the product and the assembly system:

*<assembly process> = f(<product>,<assembly system>)*

The author has extended this by introducing new functions, which capture the concurrency of the development of the product, the process and the resources.

Assembly resources are a function of the assembly process and the product; and the product is a function of the processes and the resources. These functions are now:

*<assembly resources> = f(<product>,<assembly process>)*

*<product > = f(<assembly process >,<assembly resources>)*

This integration of the manufacturing information in the Product Model is important for DFA because it solves one of its weak points, identified by Molloy (Molloy *et al.*, 1991) as the lack of consideration of the manufacturing concerns of the user. It also helps to increase the level of integration between design and manufacturing by providing mechanisms for capturing manufacturing rules and decisions.

The relations found between the Product Model and the Manufacturing Model, and the classes created for supporting them are presented in separate sections, as follows: Type of Product and Manufacturing Model, Characteristics and Manufacturing Model, and Views and Manufacturing Model.

### 7.4.1. Relations between the Type of Product class and the Manufacturing Model

According to the structure proposed for the Product Model, products can be original, variants or standard. These products can be components, subassemblies or parents. All of these categories and levels influence the kind of manufacturing facilities that will be required for assembling them.

If the product that is being designed is a Variant of an existing product, the Resources, Processes and Strategies used for the existing product are very likely to be the same that will be used in the new product. In this case, the existing information from the Manufacturing Model has to be shown to the developer.

When an Original Product is being designed, it is necessary to consider the existing processes that are available. Those processes provide the designer with the current restrictions of the system and some specifications for the product.

The designer will decide if the design will be adapted to the current resources or new processes or combinations of them are necessary for the product.

The product structure (Parent Assembly, Subassembly, and Component) influences the strategies that will be used in its manufacture. The strategy existent for similar products can give some guidelines to the designer of the product.

### 7.4.2. Characteristics and Manufacturing Model

Figure 7.13 shows the relations existent between the Characteristics class in the Product Model with the Resources, Processes, and Strategies classes in the Manufacturing Model. The key relations are explained below.

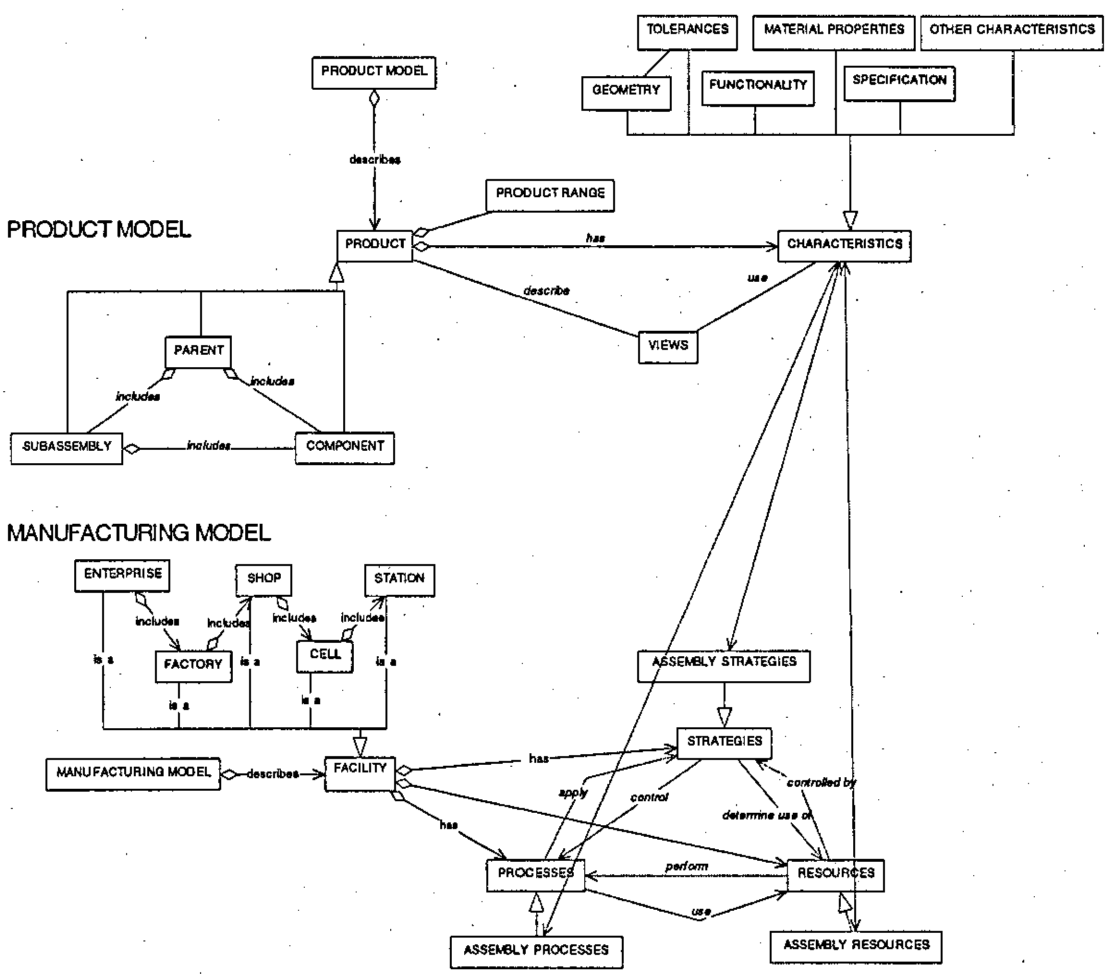



Figure 7.13 Relationship between Product Characteristics and Manufacturing Model

The Specification characteristic holds the information about the requirements that the product fulfils. The influence that the Specification has over the Resources is mainly in the selection of the equipment and tools that can match the required specifications, such as the use of hygienic materials if the product will be used in the pharmaceutical or alimentary industries.

The Geometry characteristic holds all the geometric information of the product, including the tolerances and surface finishes. These two characteristics, plus the main dimensions of the items, are necessary for the selection of the resources that are needed for producing them. So, the relation between the geometry and the resources, processes and strategies in both directions influences the selection of equipment or the adaptation of the design to the resources available.

The Material Properties class holds the material properties of the product. Depending on these properties some parameters for the processes are defined, they not only define if a part can be eliminated but also they define the joining method that can be used and possible problems during the assembly, for example with fragile or magnetic parts.

### 7.4.3. Views and Manufacturing Model

Some of the most significant relationships between the Product Model and the Manufacturing Model take place between the Product Model Views and the Resources, Processes and Strategies. The Views classes were explained in section 7.2 and illustrated in Figure 7.4, Figure 7.5, and Figure 7.6.

The difference between the Manufacturing View class in the Product Model and the Manufacturing Model information is that the first holds manufacturing information particular to a product, whilst the latter contains generic information about the facilities of an enterprise. Both are closely related, because the product's manufacturing information depends on the facilities available, contained in the Manufacturing Model. The relationships between the Manufacturing View of the Product Model and the Resources, Processes and Strategies of the Manufacturing Model are explained below.

The Manufacturing View is defined by using information from the Product Model Characteristics and from the Manufacturing Model Resources, as illustrated in Figure 7.14. This view obtains information from the Characteristics about the product to be assembled, including tolerances to be achieved, material properties, and geometric properties; this Characteristics information is used for selecting the Resources to be used for assembling the product. The information obtained from the Manufacturing Model Resources consists on the actual Resources available in the company to perform the assembly.

The relation between the Manufacturing View and the Manufacturing Model Resources is important because the product's manufacture depends on the resources available. In order to work in a concurrent environment through the different stages of the design of the product, the designer has to be aware of the resources available, as well as the developer of the manufacturing system has to know the requirements of the product. In this relationship, the Manufacturing View holds an identifier of the Manufacturing Model's Resources. It was decided to hold an identifier of the Resources, instead of copying the information, so the most updated information about the facilities can be accessed.

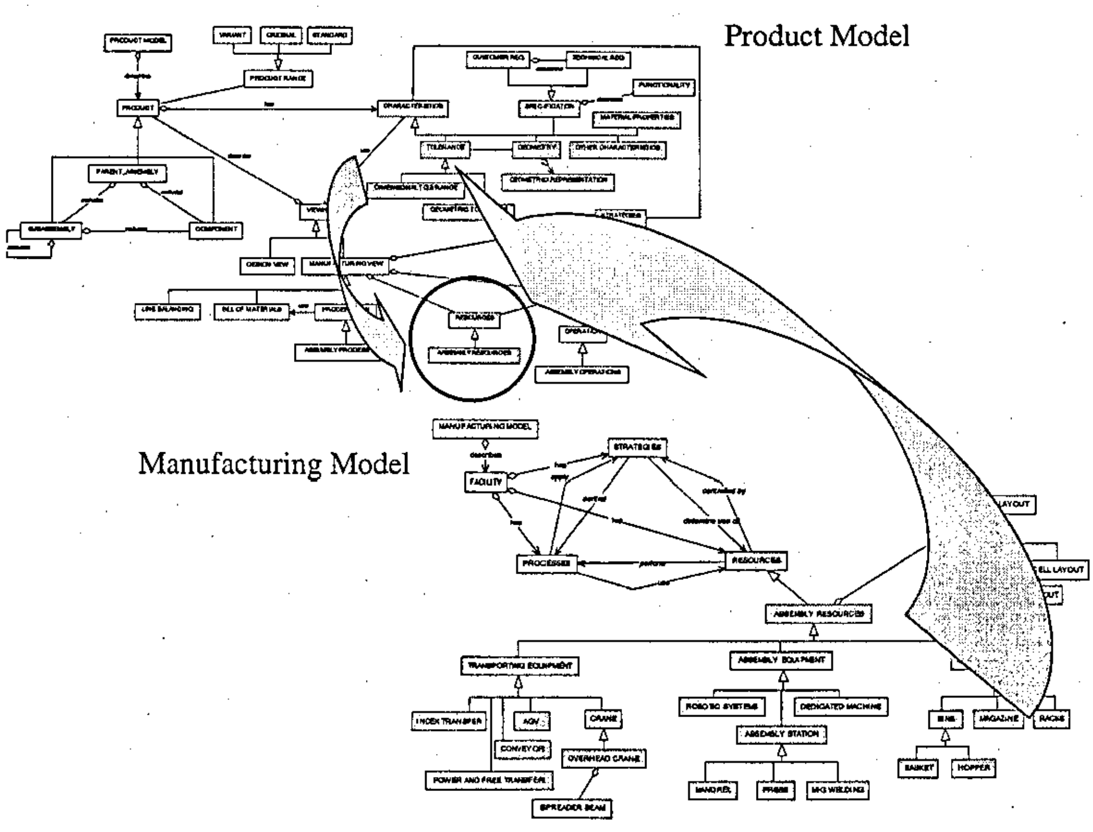



Figure 7.14 Relation between the Manufacturing View and Manufacturing Model Resources

The relation between the Manufacturing View information with the Manufacturing Model Processes is very strong, due to the fact that the manufacturing information contained in the Product Model is about specific processes required for the manufacture of the specific product. This relationship allows the development team to define concurrently both the processes required for the manufacture and assembly of the product, and the development of processes according to the requirements of the product. This relationship is depicted in Figure 7.15, in it the arrows show the flow of information from the Characteristics in the Product Model to provide information to the Manufacturing View to select Processes from the Manufacturing Model.

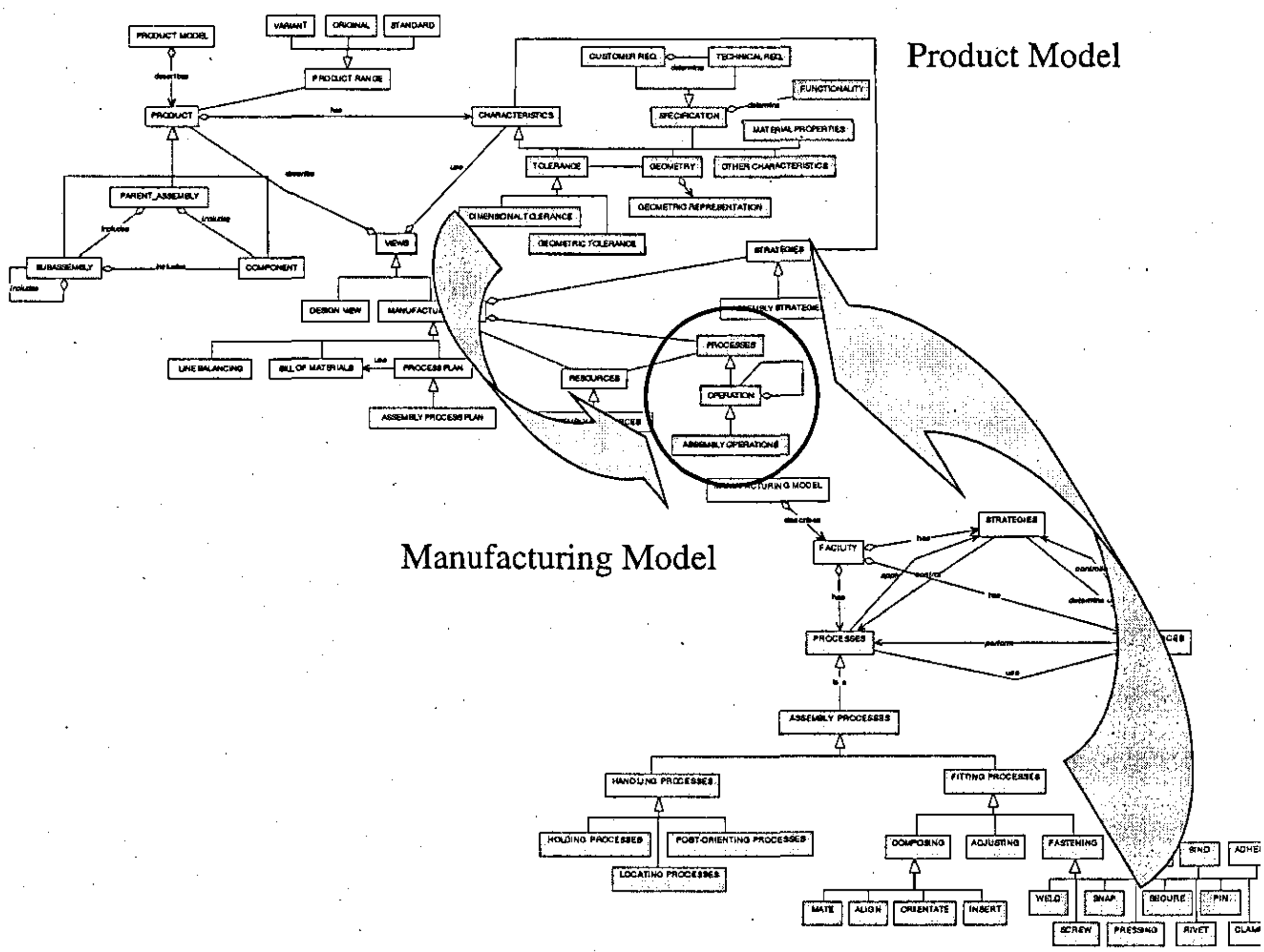



Figure 7.15 Relation between the Manufacturing View and Manufacturing Model Processes

Generic manufacturing strategies provide guidelines for the use of the resources for performing the required processes. These strategies are independent to the product, so they can be determined only with information from the Manufacturing Model. However, the author considers that assembly strategies are highly product-dependant, so information about the product to be assembled is necessary for creating them. Assembly strategies are based on the Characteristics of the product and on the generic Strategies defined in the Manufacturing Model. These interactions are shown with arrows between classes in Figure 7.16.

An example of a generic Assembly Strategy is the selection of the method of storage according to factors such as the delivery and collection patterns (bulk, flat arrangements, orientated in line), processing speed, and storage space requirements. These factors can be applied to a wide range of components in order to select the Resources and Processes. For example, if the parts require being stored in flat arrangements, then palette magazines would be the first choice. If the parts can be stored in bulk, then a bin could be recommended. In a third case if the parts require being retrieved from bulk and stored in a magazine, then an orientation process and orienting resources can be chosen according to the general characteristics of the product, such as a vibratory bowl feeder for parts with features that allow their automatic orientation.

However, there are some assembly strategies that can not be generalised and are highly product dependent. An example of product-dependant strategies is the strategies for locating, inserting, and fitting the components or subassemblies. Although some general rules are given by DFA, such as the recommendation for a top-down layer assembly, they can not be considered as strategies but only as recommendations. Even the sequence in which the parts have to be assembled depend on the specific characteristics of the product and can not be generalised, that is the reason for the great variety of approaches for Assembly Sequence Generation, explained in section 2.4.2.1.

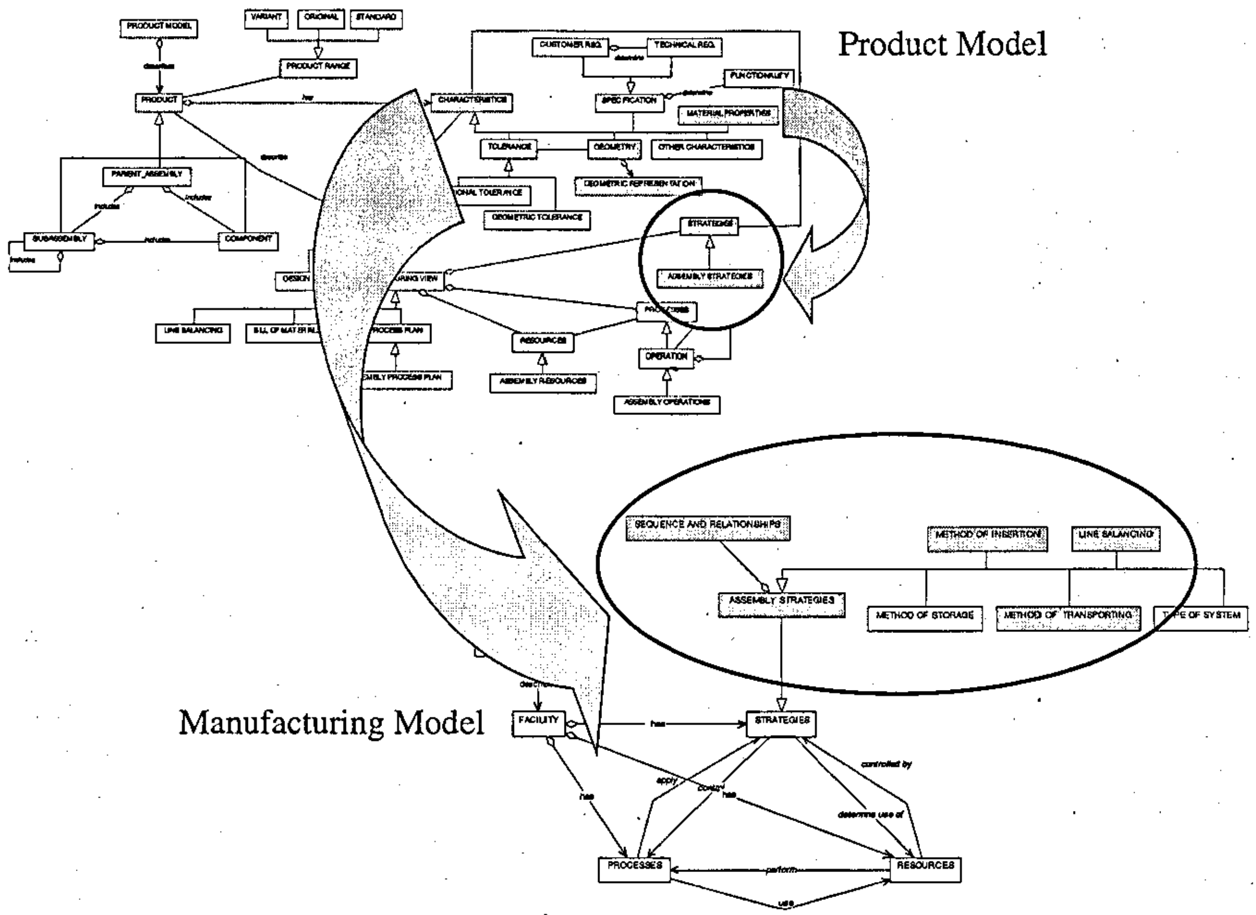



Figure 7.16 Creation of product dependant Strategies

In order to define the kind of system that will be used for assembling the product, it is necessary to know certain parameters of the product, such as the estimated production rate, life of product, number of parts, size, and weight. These characteristics have to be compared with the Strategies defined for different kind of systems and their capabilities.

Figure 7.17 shows the interaction existent between the Manufacturing Model Strategies information and the Product Model Characteristics for the Kind of System to be used for assembly, which was shown in detail in Figure 7.12.

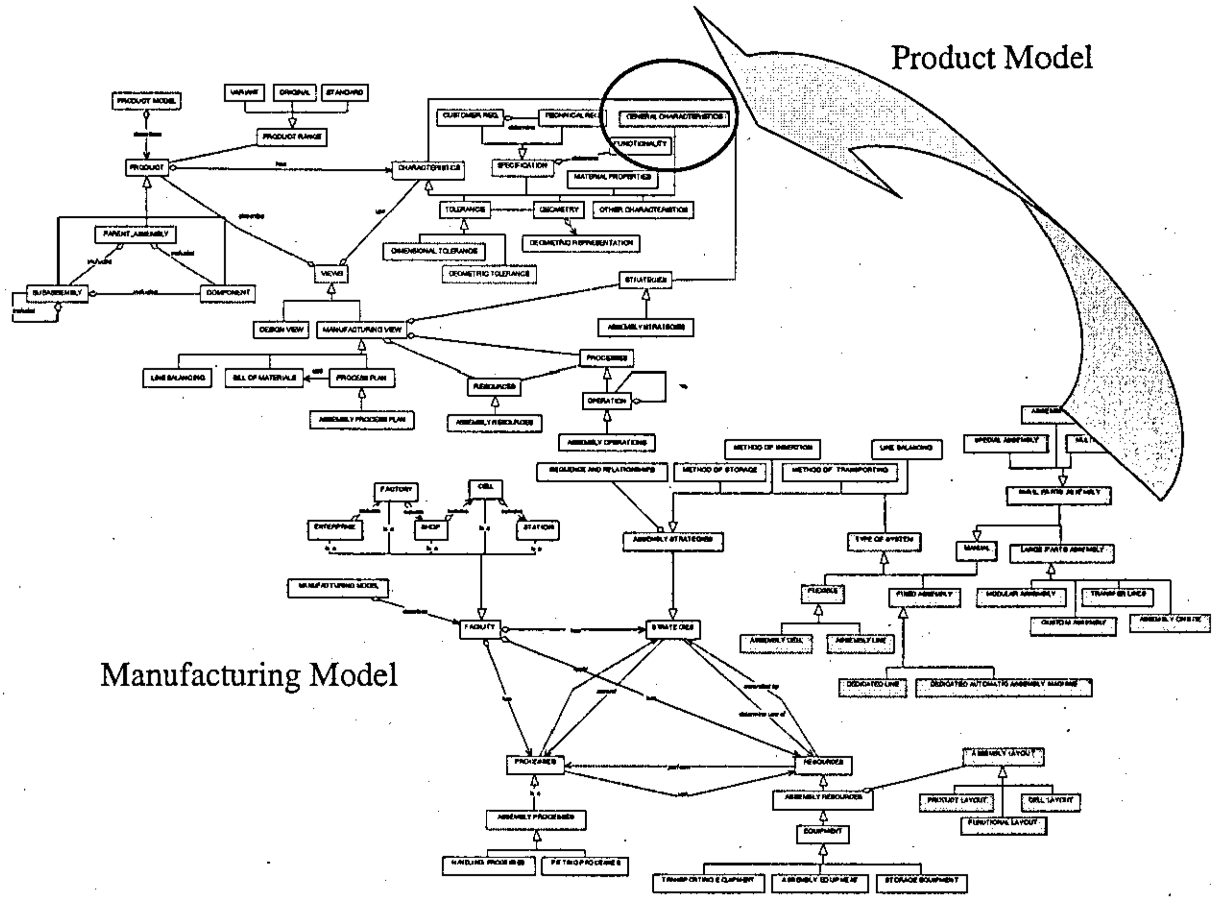



Figure 7.17 Strategies for select the type of system

## 7.5. Summary

This chapter presented the resulting structures for the Product Model and the Manufacturing Model. These structures were designed to provide information support to the assembly-related activities DFA and APP through the concurrent development process of the Product, Resources and Processes.

Multiple Views were introduced in the Product Model, and the classes for the Assembly Design View and the Manufacturing View were explained. This allows the access to the product information from different perspectives, allowing a better understanding of the modelled product.

The structures necessary for establishing a link between the Product Model and the Manufacturing Model were explained, this link allows the work in a concurrent environment, by providing the development team with information about the design of the product, design of the process, and design and selection of the resources.

Chapter 8 explains the use of the information structures defined for the information models in the construction of a Product Model, a Manufacturing Model, and DFA and APP software for evaluating and testing the ideas of this research. The results of the evaluation are presented in chapter 9.

# 8. DESIGN AND IMPLEMENTATION OF THE CASE STUDY ENVIRONMENT

## 8.1. Introduction

This chapter explains the design and implementation of the case study system produced by the author to evaluate the representation of assembly-related information in the Product and Manufacturing Models. This evaluation is done by analysing to which extent the mentioned models can provide a common source and repository of information to support interacting Design For Assembly and Assembly Process Planning applications through a concurrent engineering development cycle.

The development environment and the implementation aims for the software are explained in Section 8.2. Sections 8.3 and 8.4 explain the information structures implemented for the Product and the Manufacturing Models, following from the structures defined in chapter 7. Section 8.5 gives details of the design of the DFA and APP software applications, explained in sections 8.6 and 8.7.

The integrated approach followed in this research for concurrently supporting the development of the product, process and resources imposed strong requirements on information interactions.

The application of the software system using real information obtained from a collaborating company, and the evaluation of the results, are explained in chapter 9.

## 8.2. Software Development Environment

The software for the case study was created to enable the confirmation that the data structures defined for the information models are able to support assembly related applications. This software consists on the Product and Manufacturing Models, and Design For Assembly and Assembly Process Planning applications.

The aims of the implementation work are listed below.

1. To show the extent to which assembly-related information can be captured in a Product Model and in a Manufacturing Model.
2. To demonstrate that the structures defined for the Product and Manufacturing Models can effectively support DFA and APP activities through a concurrent development cycle. The applications defined in order to explore the use of the information models were:
- Design For Assembly application to provide the designer with guidelines through the different stages of the product design cycle, emphasising the early stages.
- Assembly Process Planning application to make recommendations through the different stages of the design and selection of Resources and Processes.
3. Support the use of shared information between the DFA and APP applications.
4. Evaluate the way in which the Product Model information influences the creation of product-dependant Strategies in the Manufacturing Model.
5. Show the information interactions between the Product Model and the Manufacturing Model.

The description of the research environment in chapter 3, included the proposed methodology and the software tools selected for the implementation of the Product and Manufacturing Models. The implementation was done using the object-oriented database Objectstore©, and the functionality of the software application was programmed using Microsoft Visual C++©.

The software system for the case study is based on the information requirements for DFA and APP evaluated in chapter 6 and in the information models explained in chapter 7.

## 8.3. Software Structure for the Product Model

The high-level information structures presented in section 7.2 were used in the testing system. The detailed aspects of this structure which are relevant to the products of the collaborating company have been implemented to provide a sufficient experimental base to prove the concepts proposed in this research.

The Product Model structure used in the testing system includes the use of Product Ranges for selecting the kind of product and its levels (Parent Assembly, Subassembly and Component). The Characteristics and Views classes were included, with the simplification of some of the subclasses, keeping those that were necessary for modelling the parts of the electrical machines used for testing the system.

In order to create the objectstore database for the Product Model, a class diagram was produced in ObjectDesign©, a software tool that assists in the creation of the Visual C++ code for creating the database classes with their attributes and relationships. The diagram used for the creation of the Product Model is included as Figure 8.1.

## 8.4. Software Structure for the Manufacturing Model

The structure defined for the Manufacturing Model in the testing software is a subset of the complete Manufacturing Model structure explained in section 7.3.

The ObjectDesign class diagram used for the creation of the Manufacturing Model is included as Figure 8.2. The structure of the Manufacturing Model used for the testing system kept the same basic structure as the generic Manufacturing Model. Some subclasses were simplified and attributes were added to some of them, in order to be able to capture the Resources, Processes and Strategies used by the collaborating company for the assembly of stators for large electrical machines.

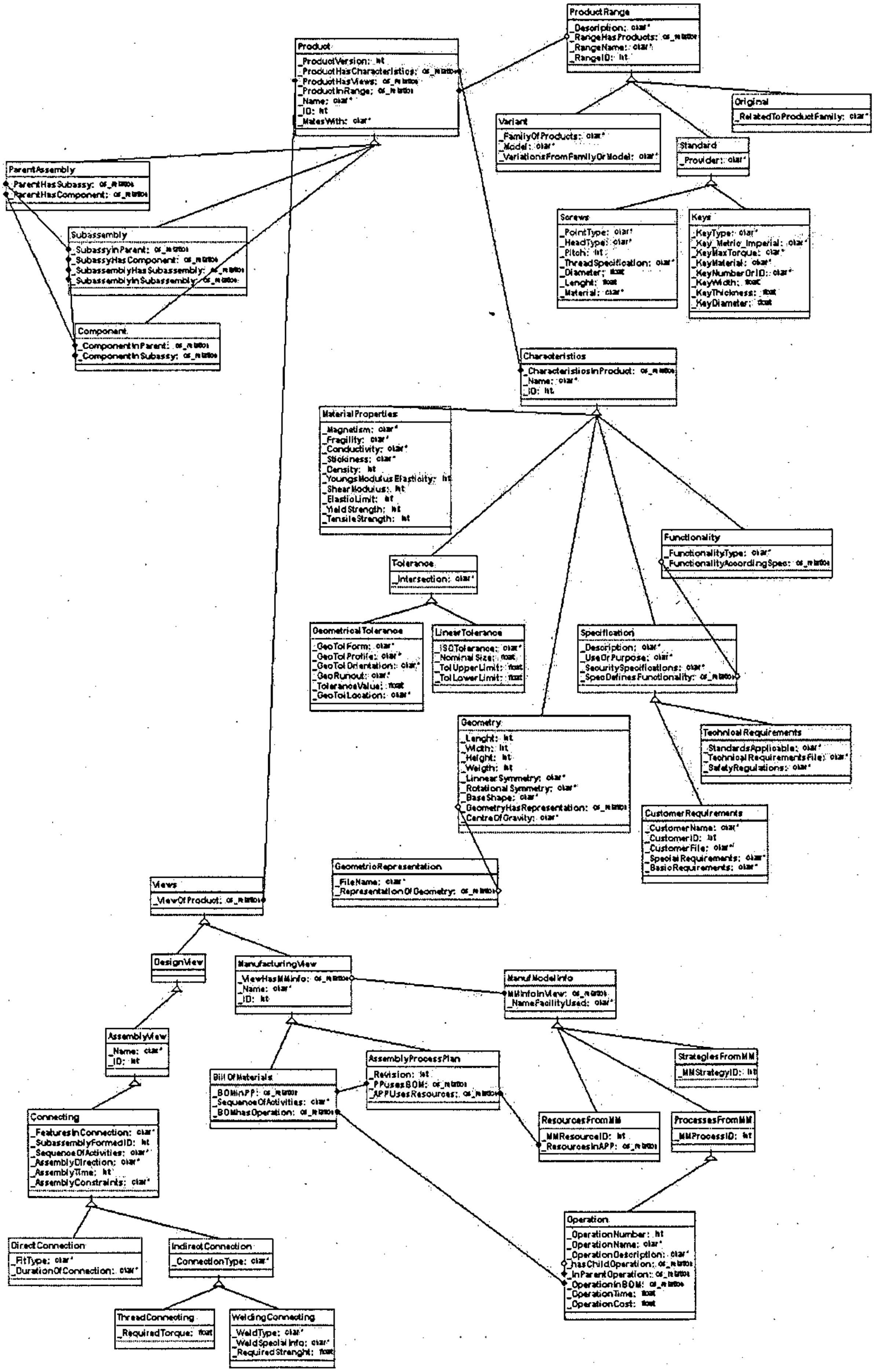



Figure 8.1 Product Model structure defined for the implementation

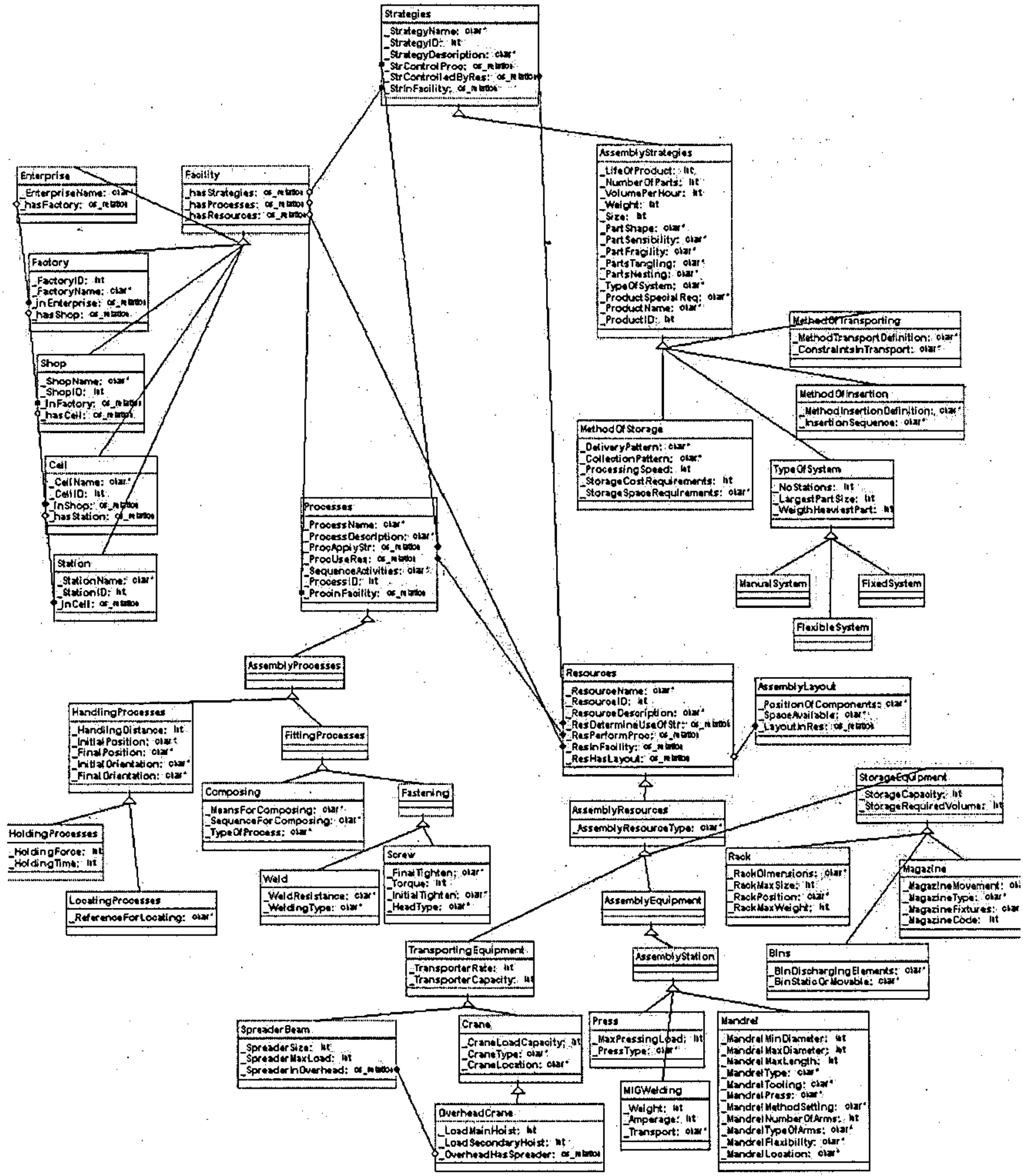



Figure 8.2 Manufacturing Model structure used for the implementation

## 8.5. Design of the Design For Assembly And Assembly Process Planning Applications

### 8.5.1. Problem Statement

The main aim of the DFA and APP software applications in the testing system is to evaluate the use of the Information Models as their source and repository of information. They were provided with limited functionality, if compared with commercial systems, but sufficient for performing basic Design For Assembly and Assembly Process Planning tasks.

The results of the influence of DFA and APP through the concurrent development cycle, presented in chapter 6, were evaluated in order to define the functionality of the DFA and APP applications. As a result of that evaluation, the information requirements of DFA and APP were defined using a requirements trace matrix, included in 8.5.2. This matrix was used as the starting point for defining the functionality for the DFA and APP applications.

The DFA and APP software applications have to be able to support the concurrent development process from the early stages. The author decided to create simple software applications rather than using commercial software, such as Boothroyd's DFMA or TeamSet, that are focused on the assemblability analysis at the detail design stage of the process. However, in the further work proposed in section 10.4, the design of interfaces with commercial software is proposed in order to be able to use their functionality at the detailed design stage using the information stored in the Product and Manufacturing Models.

### 8.5.2. Requirements Trace Matrix

The requirements trace matrix captures the requirements that must be satisfied by the system. These matrixes are used in the methodology proposed by Texel and Williams (Texel and Williams, 1997) for the requirements engineering stage of the UML Use Cases analysis. The system requirements are established by analysing *'shall'* statements obtained mainly from the methodology used in this research.

The requirements trace matrixes for DFA and APP are presented in the following sections, indicating the stage of the development cycle in which they are used. The system performs some of the activities listed and provides assistance to the user for performing other activities.

#### 8.5.2.1. Requirements Trace Matrix For Design For Assembly

1. The user shall determine what is being designed, the product, the process, the resources or a combination of them
2. The system shall check for information available about the product, processes and resources
3. The system shall ask for the estimated volume of products to be assembled, the estimated life of the product, and determine the possible number of parts in the assembly
4. The system shall suggest the kind of assembly system for the product, and classify it as manual, automatic with specialised machines, flexible in cells, or lines using robots.
5. The system shall identify the assembly resources available.
6. If it is not possible to determine the kind of system to be used, due to lack of information, the system shall consider it as automatic.

CONCEPTUAL DESIGN OF THE PRODUCT

7. The system shall search for products that fulfil the same or similar functions
8. The system shall ask the user if the design will be of a new product or a existing one.
9. Search for existing modular components that could be used in the design
10. The user shall identify similar items in order to highlight in them the relevant handling and locating features.
11. The system shall advise the designer about possible problems with parts that are slippery, delicate, flexible, very small or very large, or hazardous to the handler.
12. The system shall highlight surfaces with high tolerances, to ensure that are functional.

13. The system shall advice the use of features to avoid jamming of parts that tend to nest or stack when stored in bulk. Avoid the use of features that will allow tangling of parts
14. The system shall make the designer aware of problems caused by parts that stick together or are slippery, delicate, flexible, very small or very large or that are hazardous to the handler.
15. The system shall encourage the use of end-to-end and rotational symmetries in the parts.
16. If the parts cannot be symmetric, the user shall make them obviously asymmetric.
17. The system shall advise that the assemblies should be done following a layer sequence in a top-down straight line.
18. The system shall make the designer aware of assembly problems caused by parts heavier that 4.5 kg, due to the inertia for accelerating the mass, moment of inertia caused by rotating the mass and the size of the part relative to its assembly clearances.

EMBODIMENT STAGE OF THE PRODUCT DESIGN

19. The system shall help the user to identify the base parts, and ensure that they are stable and the other parts can be easily inserted in an upside-down straight line.
20. For individual components, the system shall detect the factors that affect manual handling: Size, thickness, nesting, tangling, frazzling, flexibility, slipperiness and stickiness.
21. For individual components, the system shall analyse the geometry of the parts, advising the use of end-to-end and rotational symmetry, or the design of fully asymmetrical parts.
22. For individual components, the system shall analyse the geometry to find possible problems of tangling or nesting.
23. For individual components, the system shall analyse the geometry and material to detect risks of sticking parts, or if the part is delicate, fragile, flexible, moisture sensitive, magnetic, sensitive to static electricity or hazardous to the handler.
24. The system shall determine if there is suitable equipment available for handling the parts.

25. The system shall advice the user about providing the parts with handling features
26. The user shall determine the centre of gravity of the part and assess its stability
27. The system shall recommend using Boothroyd and Lucas tables for handling.
28. The user shall classify the parts according to the alpha and beta symmetries and determine if can be manipulated with one hand, two hands or need mechanical help.
29. For assemblies and subassemblies, the system shall propose the reduction on the number of parts.
30. For analysing the possibility of eliminating a part, the system shall check the following:
    - If the part moves relatively to other parts during operation of the final product.
    - If the part must be of a different material than, or must be isolated from, other parts.
    - If the part must be separated from other parts in order to allow the assembly or disassembly of other parts.
    - If the part must be separable for maintenance or replacement

31. For assemblies and subassemblies, the system shall identify tight tolerances and surface finishes that make insertion difficult
32. For assemblies and subassemblies, the system shall check if the assembly process follows a pyramid sequence, inserting the parts in a straight up-down movement and in layer sequence.
33. For assemblies and subassemblies, the system shall check that there is a stable base part on which to build the assembly
34. For assemblies and subassemblies, the system shall check the kind of fasteners that are being used, proposing the change to assembly friendly fasteners, in the following order: snap-fit, plastic bending, riveting and screwing.
35. When screws are compulsory, the system shall advice the use the following sequence of preference for the points, (the first is the best): Oval point, cone point, dog point, chamfer point, header point, rolled thread point.
36. For assemblies and subassemblies, the system shall recommend that the parts have chamfers and other guides for assembly

37. The system shall recommend the use of the Boothroyd table for Manual Insertion and Lucas' fitting analysis.

DETAIL STAGE OF THE PRODUCT DESIGN

38. In the Detail Design stage, the system shall perform a final evaluation of the design, applying the same steps as in the conceptual stage.
39. The system shall advise on the use of the standardisation of components such as screws.
40. The system shall evaluate the handling and insertion processes proposed and the equipment selected for the task.

#### 8.5.2.2. Requirements Trace Matrix for Assembly Process Planning

FOR SUPPORTING THE DESIGN OF PROCESSES

1. The system shall check if there is information available of the product, and resources that will be used to assemble it.
2. The system shall check constraints about the processes to be used, including the type of system (the selection method is proposed at the beginning of the product design).
3. If the resources are defined, the system shall help the user to analyse their capabilities and limitations
4. If neither the product nor the resources are known, the system shall show a list of current processes to guide the designer in the decision taking process
5. The system shall analyse the relation between parts, taking into account the level of completeness and development of the information, if the product information is in the conceptual stage, it is very likely that it will change completely.
6. The system shall assist the user to build an assembly tree representation or a BOM for Assembly, it is necessary for the assembly plan creation
7. The system shall check for the definition of the base part upon which the assembly will be done.
8. The user shall define actions that have to be performed according to the assembly tree

9. The system shall provide guidelines to select the processes, according to the product characteristics, the recommended equipment, and the actual equipment available
10. The system shall help the user to define the sequence for performing the assembly, according to processes.
11. The system shall produce a precedence diagram for the assembly sequence

FOR SUPPORTING THE DESIGN OF RESOURCES:

1. The system shall check if there is information available of the product and its related processes, and its stage of development
2. The system shall perform the analysis of the assembly processes required for assembling the product
3. The system shall apply APP considerations to help the user to select the principles that will be followed, evaluating against economic, process and systems constraints
4. The system shall help the user to group the equipment in cells and stations
5. The system shall assist the user in classifying the resources to be used according to the BOM.

## 8.6. Design For Assembly Application

The originality of the research is concerned with the information model structures. A rudimentary Design For Assembly application has been constructed to test the information structures.

The Design For Assembly application developed for the testing system is intended to be a system that provides information to the development team at the moment that the decisions are taken and not to be used as a separate evaluation tool. In order to achieve this, the DFA application runs in the background of the software application designed for inserting and retrieving the information of the Product and Manufacturing Models.

When a parent assembly or subassembly is being introduced to the system, general information is requested, including estimates of the amount of parts that will constitute the product, the expected life of the product, and the expected general dimensions. This information is used to recommend the kind of system to be used for the assembly, as shown in Figure 8.3. At the conceptual design stage, only some information of the product is available. Using that information the system can provide the designer with information about the kind of systems that could be used for assembling the product by giving estimates of what is intended to obtain at the end of the product design cycle.

Geometry is one of the most clearly needed characteristics for defining a product. In the testing system, the general geometric characteristics have to be introduced manually into the system. Research work, reported in chapter 2, have successfully explored the way for automatically capturing geometric information from a CAD system and introduced it into a Product Model, as shown in Figure 8.4. The information structures proposed in this research can be extended to hold all the detailed geometric information obtained in that way, however this was out of the scope of this research.

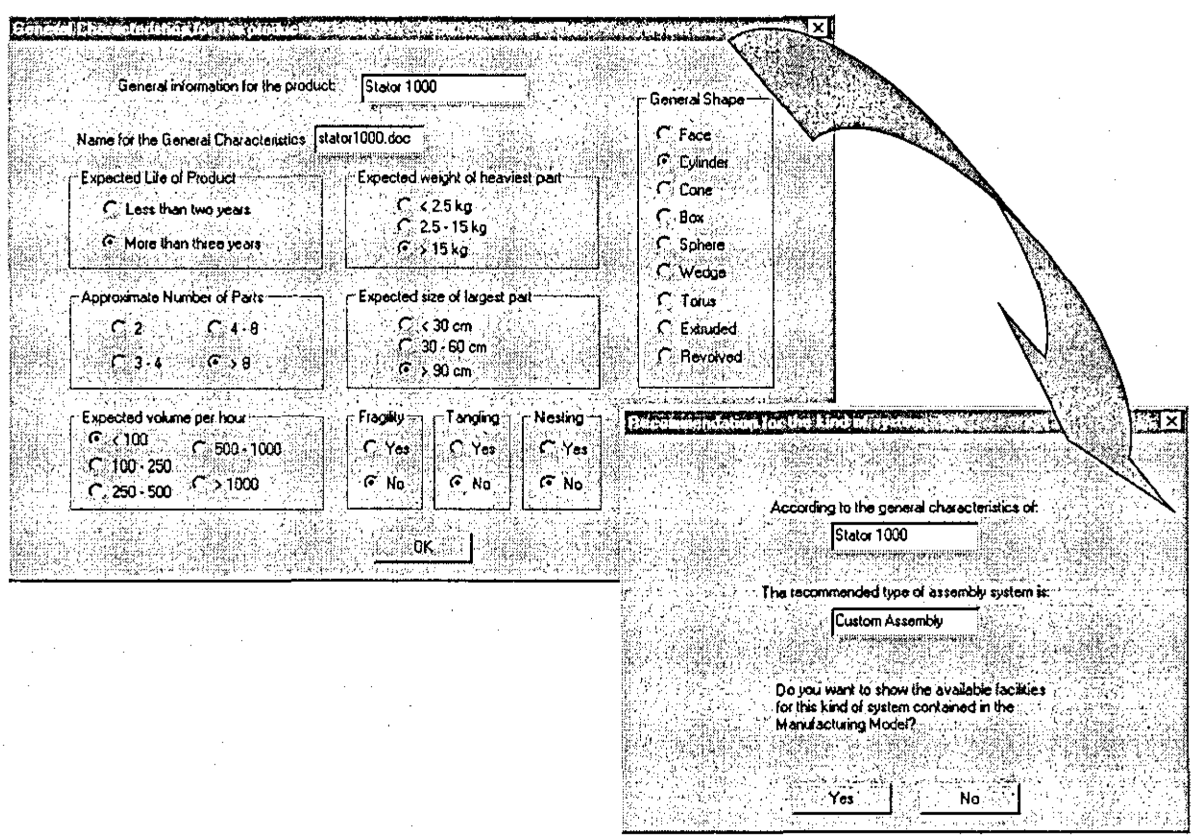



Figure 8.3 General information about the part and recommendation of system

The geometry that the testing system can hold is limited to the general dimensions of the part, but for providing complete geometric information to the user, a link was created with a graphics programme that can display the drawings of the part.

When the geometry is inserted in the Product Model, the symmetry of the part is introduced using Boothroyd's notation (Boothroyd *et al.*, 1994), as shown in Figure 8.4. This is to capture the information necessary for performing a Boothroyd's DFMA analysis in the detailed design stage.

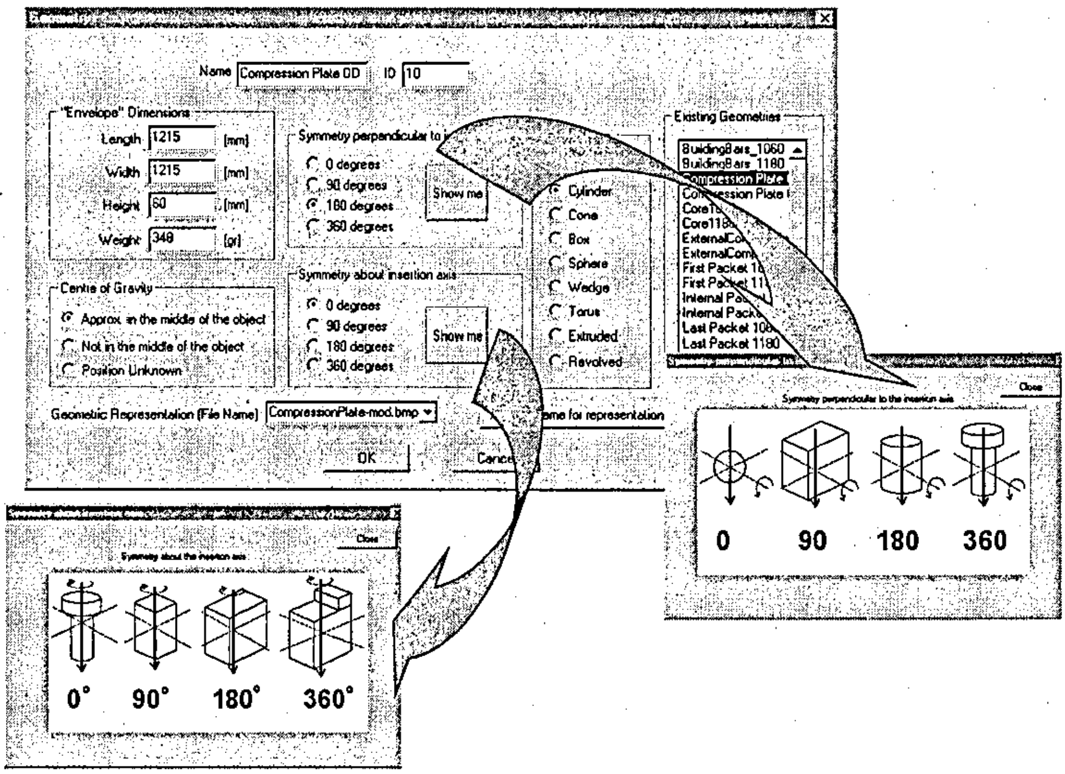


Figure 8.4 Inclusion of symmetry in the capture of Geometry

When the material for the part is selected, the user receives information about the stickiness, magnetism, fragility, and conductivity, so the designer will immediately know about possible problems in the assembly caused by the kind of material selected, as shown in Figure 8.5. New material definitions can be introduced into the Product Model by providing the name, density and attributes of the material. The information contained in the definition of the material can be extended in order to support other applications that require other attributes.

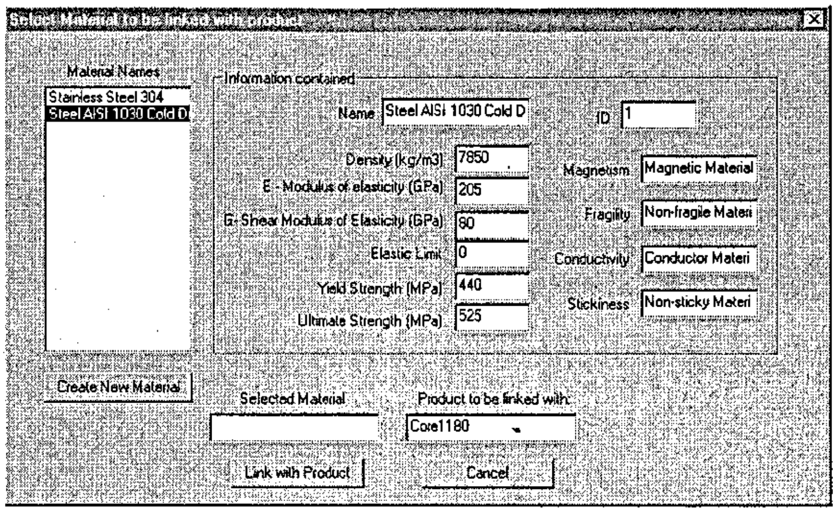



Figure 8.5 Dialog box for selecting material

When the characteristics of the part have been introduced, the DFA application asks four questions, explained in section 6.2.3.2, to the user of the system in order to determine if the part is essential or can be eliminated. These questions are asked every time the user links characteristics to the part, so it is used in the different stages of the product development cycle, giving the possibility of changing the result of the questions when more information is available. These questions are shown in Figure 8.6.

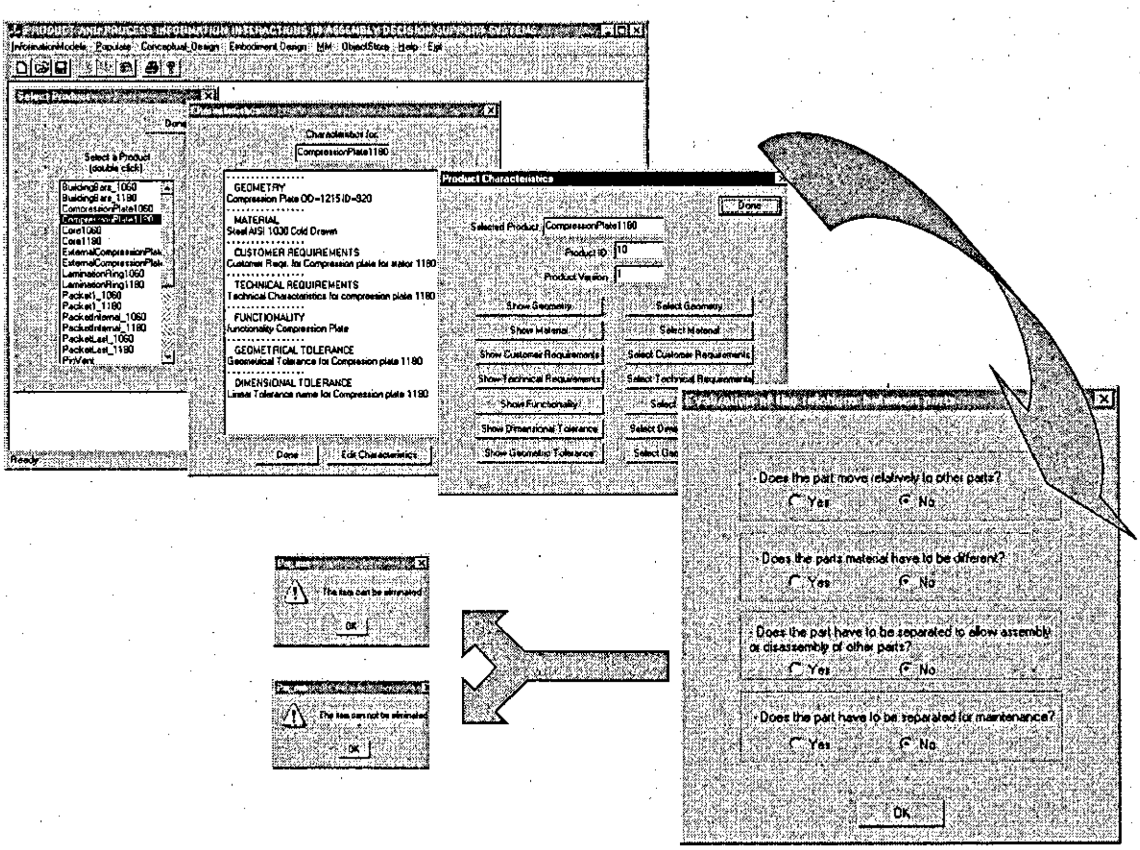



Figure 8.6 Evaluation of the possibility for eliminating the part

The DFA application evaluates the Characteristics information of the part, including Geometry, Material, and Tolerances, and provides recommendations if that information contains something that could cause a problem in an assembly operation, as shown in Figure 8.7.

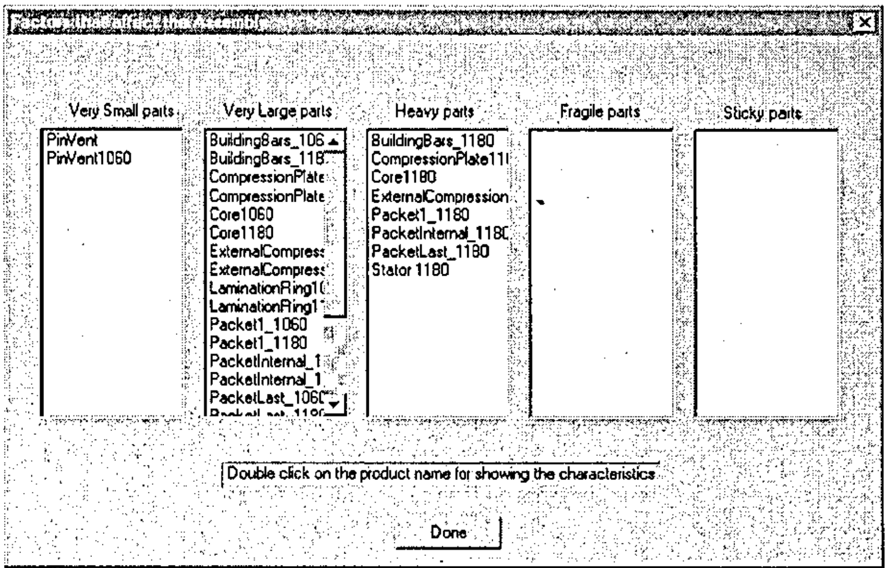



Figure 8.7 Factors that affect assembly – Individual parts

When information about several components has been introduced, the links between them have to be defined. The kind of connections included can be a direct, welded, or threaded connection. Figure 8.8 shows the dialog box in which direct connection information is entered, including information about the direction of insertion, its duration, the fit type, and if it needs external help to maintain the orientation of the part.

After creating links between the parts, the system generates an assembly tree in which the parent assembly with the subassemblies and components are included, showing their assembly sequence number. DFA evaluates possible insertion problems using the information stored in the Product Model, and finds possible assembly problems, both for individual parts and in assemblies, as shown in Figure 8.7 and Figure 8.9.

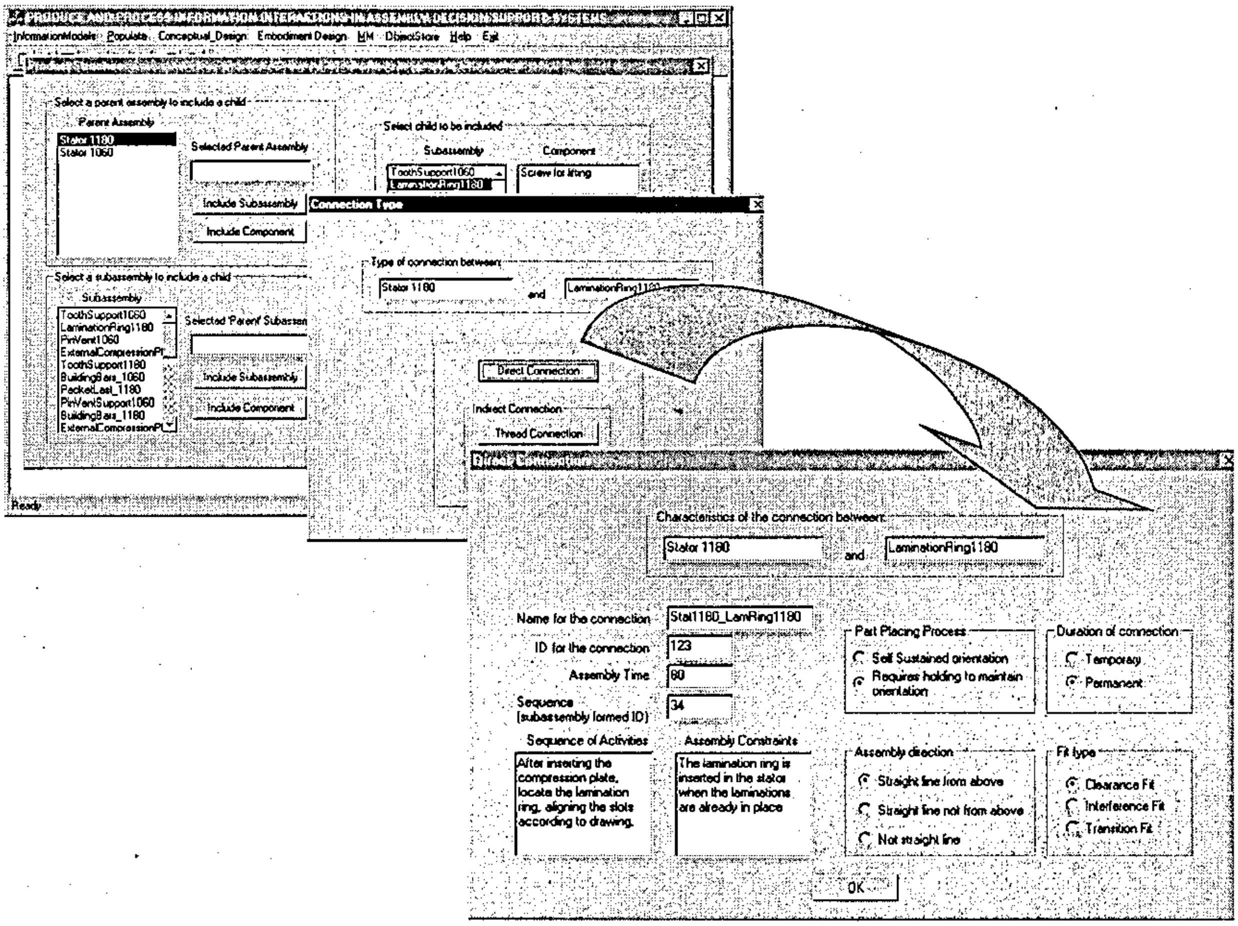



Figure 8.8 Capturing information for the connection between parts

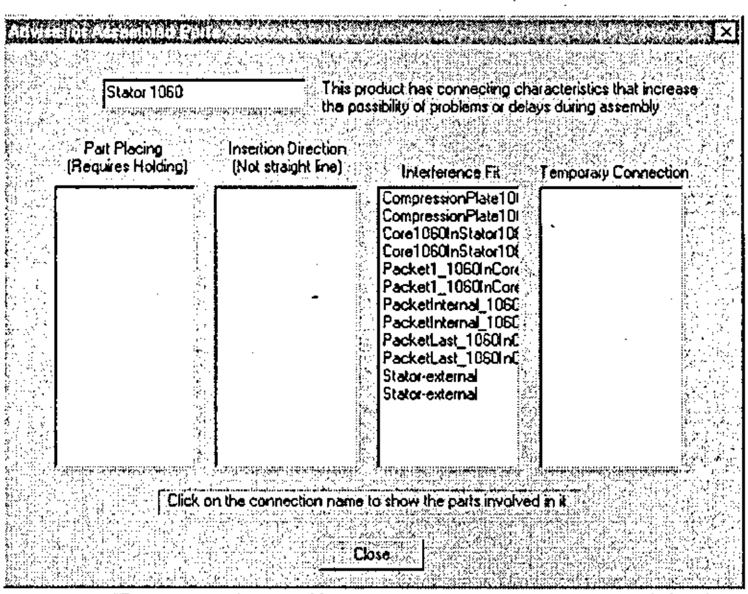



Figure 8.9 Factors that affect assembly – Assembled parts

The functionality included in the DFA application shows the usability of the Product Model and Manufacturing Model for supporting DFA through the concurrent development cycle.

## 8.7. Assembly Process Planning Application

The originality of the research is concerned with the information model structures. A rudimentary Assembly Process Planning application has been constructed to test the information structures.

The Assembly Process Planning application follows the same approach as the DFA application, by running in the background of the software application designed for inserting and retrieving the information of the Product and Manufacturing Models.

The type of system to be used for the assembly is selected at the beginning of the concurrent development cycle, according to the general characteristics of the product, as shown in Figure 8.3. Using the system's recommendation, the available facilities information in the Manufacturing Model is shown to the user, as depicted in Figure 8.10.

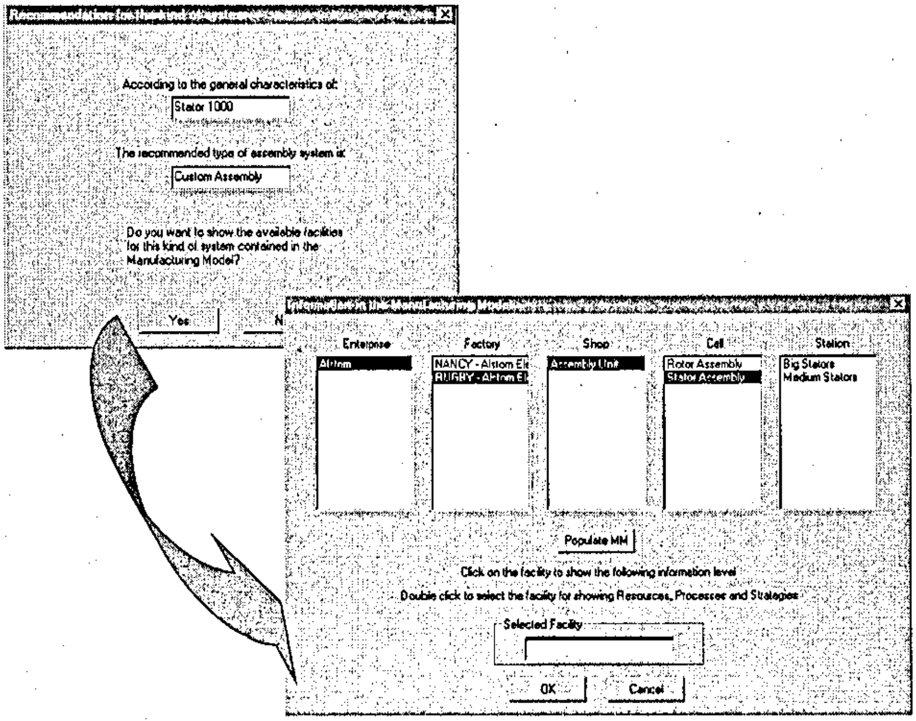



Figure 8.10 Information available in the Manufacturing Model

When the product design cycle reaches the stage of embodiment design, the information in the Product Model includes connection between parts. Using that information, a product assembly tree is built, showing not only the structure of the product, but also the sequence for assembly; this is shown in Figure 8.11.

Due to the fact that this information is constructed in parallel with the design and selection of the facilities to be used, it is likely that this information can change, so it is possible to rebuild the structure of the product, according to the changes in connections between parts.

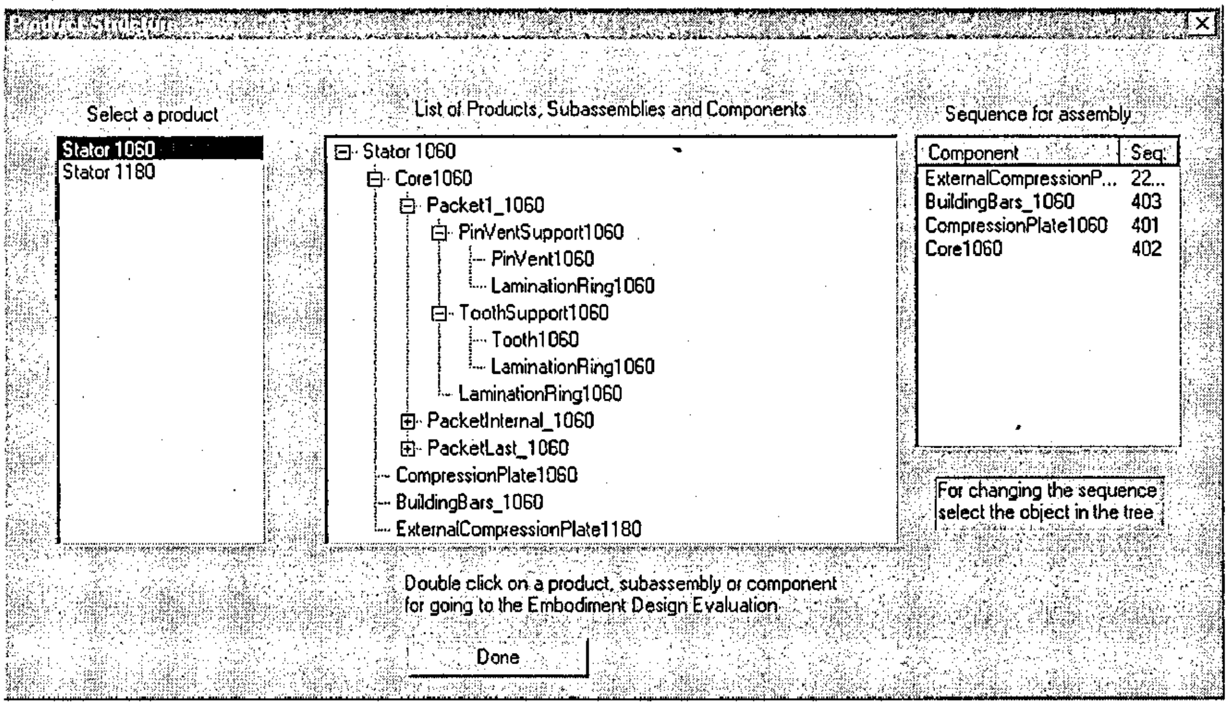



Figure 8.11 Product structure represented as an assembly tree

APP provides advice about which Resources and Processes can be used for the assembly operations required, according to the information contained in the Manufacturing Model. In order to do this, the system analyses which Resources and Processes comply with certain key characteristics of the part, such as its size and weight. This analysis is performed internally by comparing the required characteristics of the part with the capabilities of the facilities. Figure 8.12 shows an example where the resources in the Manufacturing Model that fulfils the requirements for specific parts are displayed. Details of the displayed facilities can be accessed in order to check their characteristics.

Using dialog boxes similar to the one shown in Figure 8.12, the user can select the resources, processes and strategies necessary for assembling the product. These are selected from the Manufacturing Model, and copied into the Manufacturing View of the Product Model.

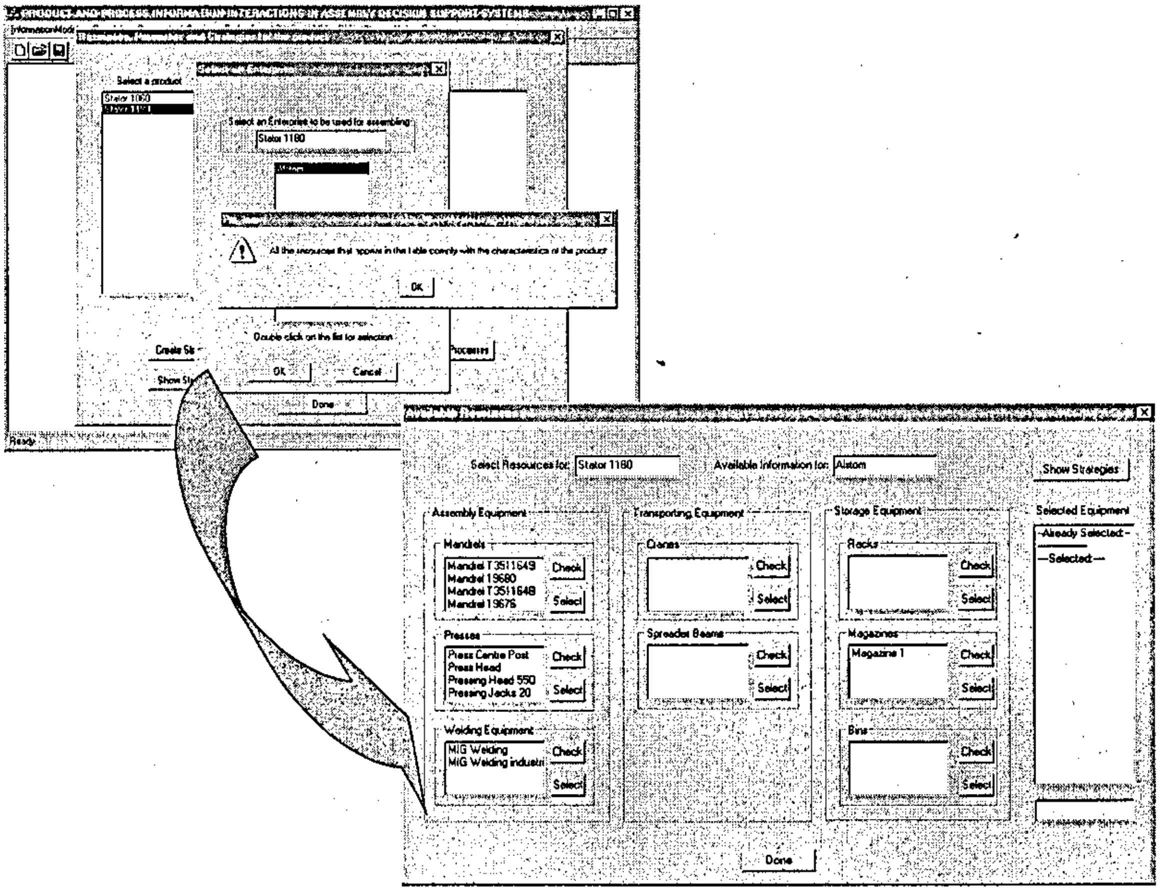


Figure 8.12 Facilities that comply with the requirements of the part

The product assembly sequences, including the selected Resources, Processes and Strategies are shown in Figure 8.13. This information can be used as the basis for constructing a detailed Bill of Materials for assembly.

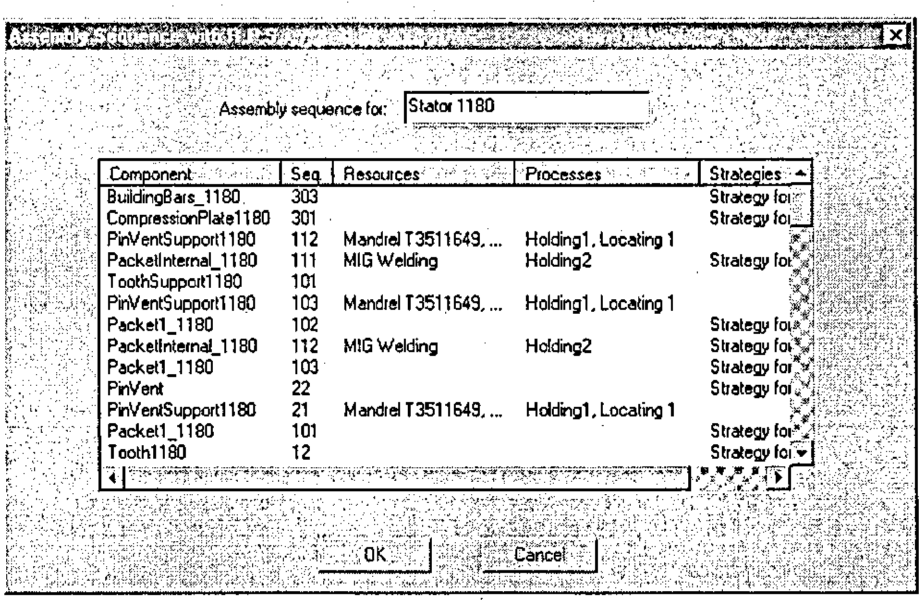



Figure 8.13 Assembly sequences including Resources, Processes and Strategies

## 8.8. Summary

This chapter presented the development of testing software to be used for demonstrating the possibility of using the Product Model and the Manufacturing Model for supporting Design For Assembly and Assembly Process Planning software applications. The functionality of this software is defined using a requirements trace matrix.

The capabilities of the software applications for DFA and APP were explained. These applications were designed to provide recommendations through the different stages of the concurrent development of the product, process and resources, emphasising in the early stages.

Chapter 9 explains the industrial example used for testing the testing software and the results obtained.

# 9. RESULTS OF THE CASE STUDY

## 9.1. Introduction

This chapter describes the tests that show how the information structures defined for the Product Model and the Manufacturing Model can capture product and facilities information to provide common support for the use of Design for Assembly and Assembly Process Planning applications. The testing software used was explained in chapter 8. Each section of this chapter consists of sets of tests, which explores the ideas developed in the research.

Section 9.3 explores the use of the structure defined for the Product Model as a source and repository of the product's assembly-related information.

Section 9.4 explores how the Manufacturing Model structure defined in the research enables Assembly Resources, Processes and Strategies to interact in order to provide information about the available assembly facilities.

Section 9.5 explores how the combined information structures of the Product Model and Manufacturing Model interact to support the selection of Resources, Processes, and Strategies for a product.

Section 9.6 explains how the structures of the Product Model and Manufacturing Model are used to integrate DFA and APP applications to provide recommendations to the user of the system through the product life cycle.

## 9.2. Introduction to the Industrial Example

In the framework of the Manufacturing Information Models project, developed at Loughborough University, and explained in section 3.2, a collaboration link was established with Alstom Electrical Machines, which manufactures large electrical machines. Access was given to information about the characteristics of stators of electric generators and the facilities available in their factory located at Rugby, England. Figure 9.1 shows the appearance of the stators used for populating the Product Model of the testing system.

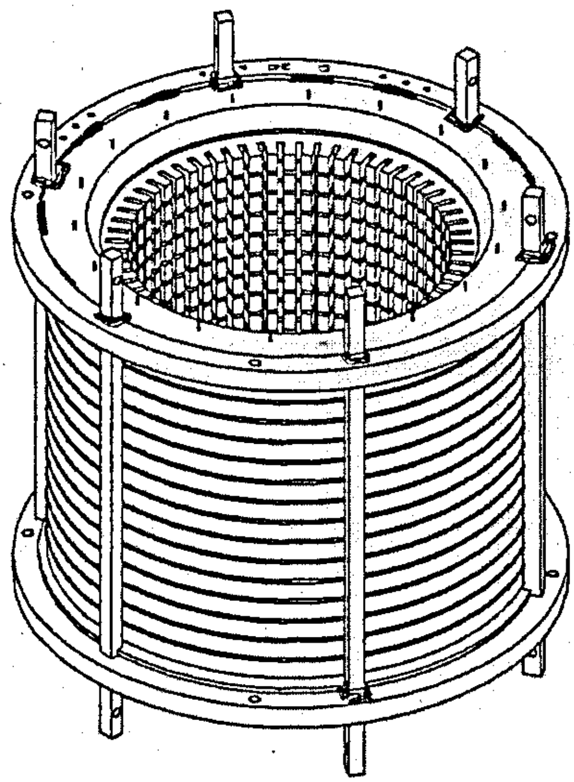


Figure 9.1 Stator of an electrical machine

Electrical stators are formed by several parts and subassemblies put together using different connecting methods. They provide a good case study for this testing system in terms of the constituent number of parts and subassemblies, the geometric, material and tolerance characteristics and the variety of equipment and facilities required for their assembly.

Stators are formed by laminations, grouped in packets. These packets are separated by pins, which provide ventilation for cooling the stator. Externally, the stator shows several bars and at the ends, it has compression plates, as shown in Figure 9.2.

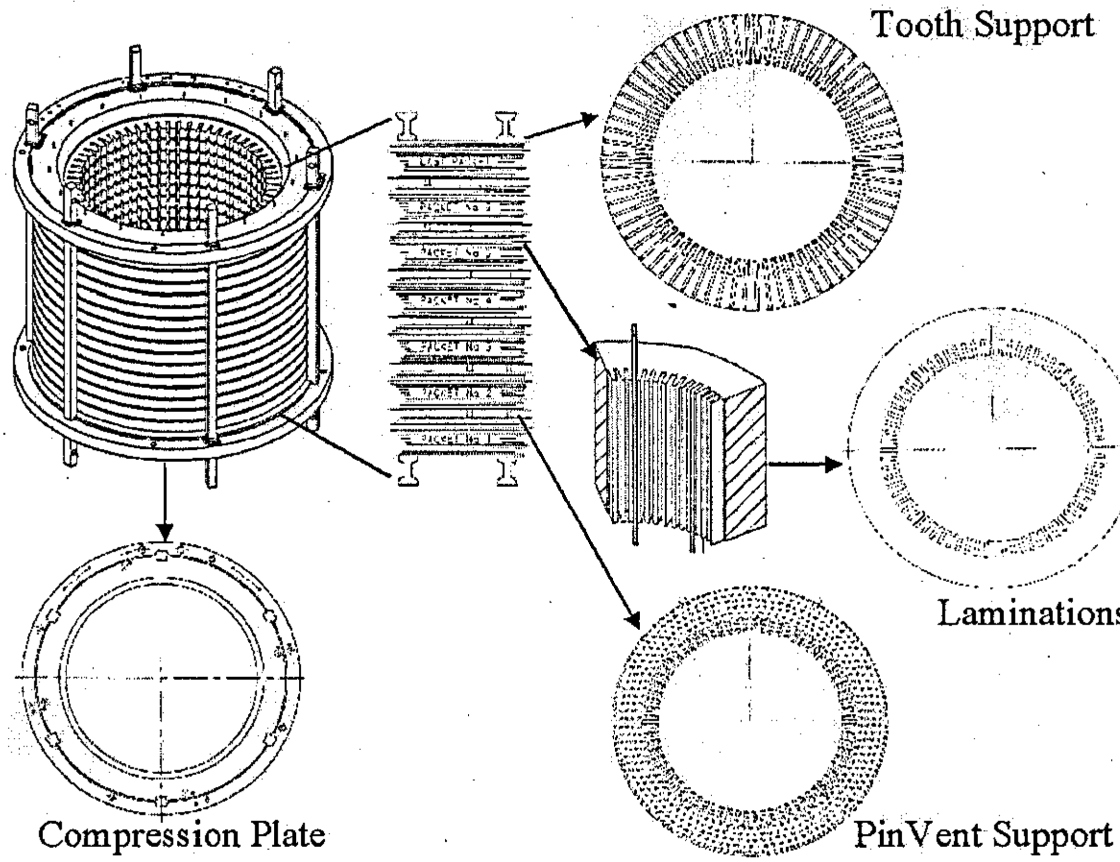



Figure 9.2 Components of an electrical stator

The facilities used for building stators are a mandrel, in which the laminations are inserted to form packets, overhead cranes for transporting the components and the completed stator, floor cranes for manipulating heavy components, presses for compressing the packets of laminations, MIG welding machines, and rotary jigs. Figure 9.3 illustrates some of the facilities available in the factory.

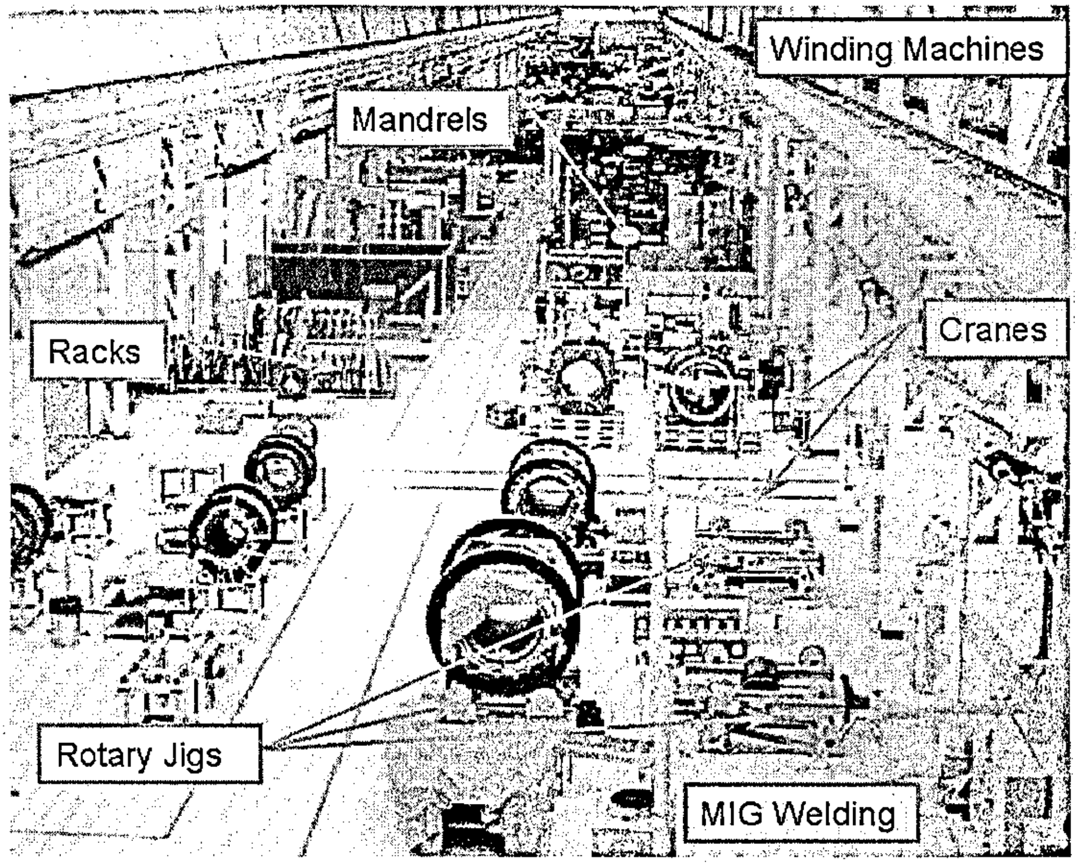



Figure 9.3 Resources used for assembling stators

## 9.3. Product Model Structure to Capture Product Assembly Information

This section reports a test aimed at the following

- To evaluate the extent to which the assembly information of a product can be captured in the Product Model.
- To show how the products can be instantiated at different levels in order to capture relationships between parts and to explain how the relations between parent assemblies, subassemblies and components are captured in the Product Model.
- To describe how the products can be classified in different product ranges in order to create families of products that help the designer in the creation of new products.

This test is divided in two sections, the first is for individual products or components and the second is for assemblies and subassemblies.

### 9.3.1. Populating individual products or components

The products can be classified in Ranges of products, namely Original, Variant, and Standard. Classifying the products in Ranges allows the designer to have easy access to the information about products that are similar to the one that is being designed, and potentially reducing the development time. The way in which Ranges were included in the structure of the Product Model, and their associations and inheritance are illustrated in Figure 9.4.

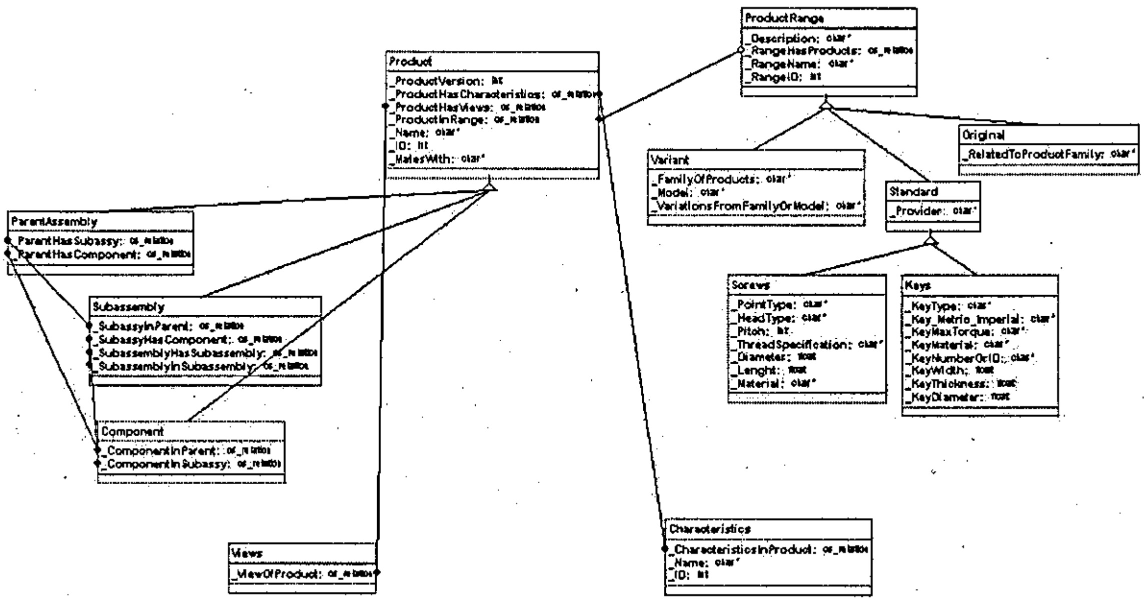



Figure 9.4 Detail of the Product Model Structure showing Ranges and levels

The products can be instantiated at different levels, including Parent Assembly, Subassembly and Component. These levels allow the capture and classification of the parts that form a complete product. Figure 9.4 illustrates these classes and their relationships in the structure of the Product Model.

The creation of the product *'Stator 1280'* as a parent assembly in the *Stator* Variant range of products is shown in Figure 9.5. This Parent Assembly will hold all the subassemblies and components that compose this product. Figure 9.5 shows that other products have already been introduced into the Product Model, most of them are Variant Products, as can be seen in the list, from which *Stator* is selected as the range for the new *Stator 1280.* The existent Parent Assemblies are shown in the list that appears at the right of the dialog box; these existent Parent Assemblies are *Stator 1060* and *Stator 1180.*

Figure 9.6 shows the relation between the information that appears in the dialog box shown in Figure 9.5 and the Product Model information structure shown in Figure 9.4.

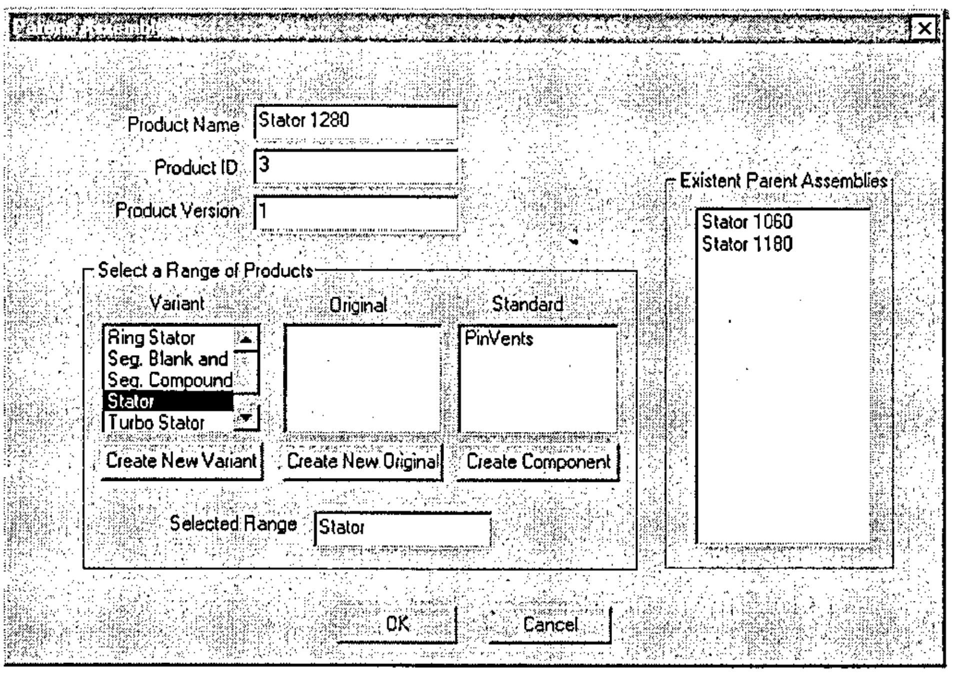



Figure 9.5 Creation of the Parent Assembly

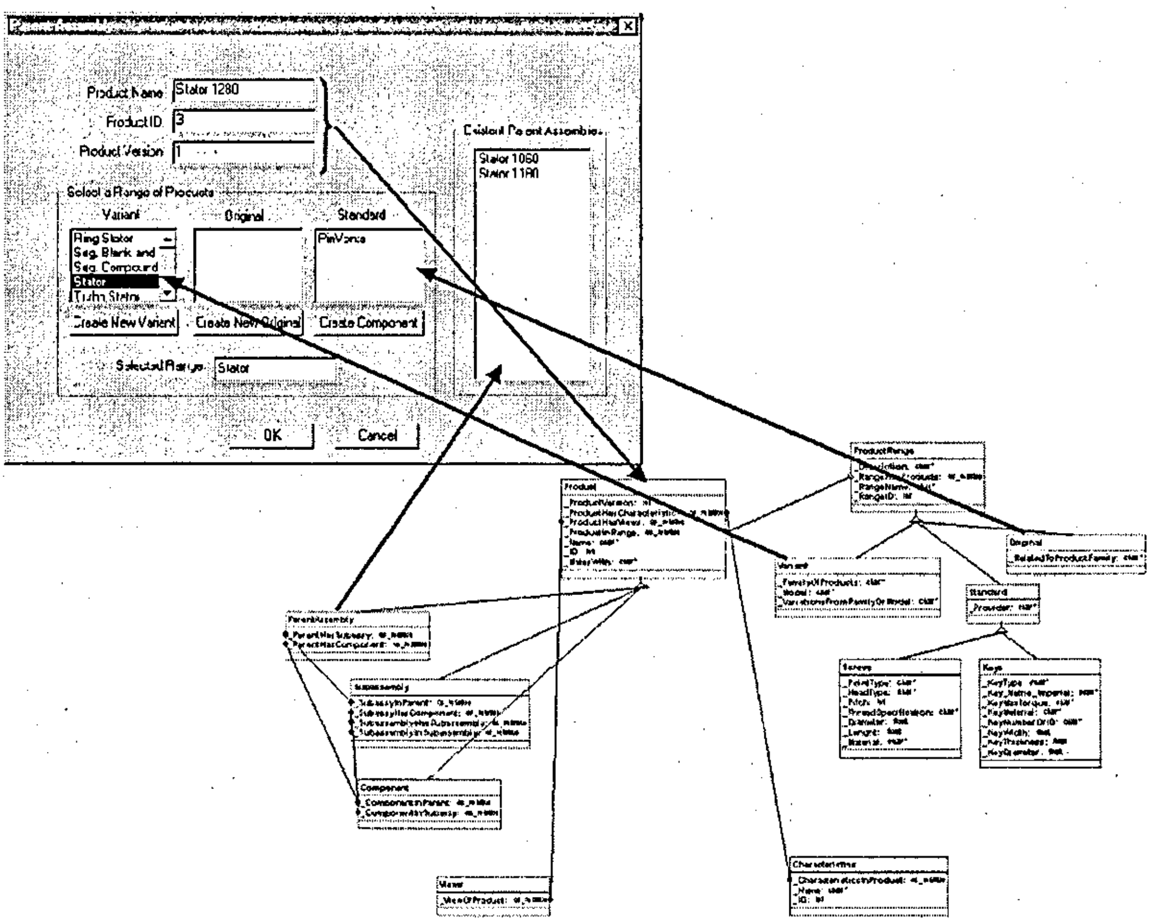


Figure 9.6 Information of the Parent Assembly in the Product Model structure

Figure 9.7 and Figure 9.8 illustrate the creation of Subassemblies and standard Components following a procedure similar to the creation of Parent Assemblies.

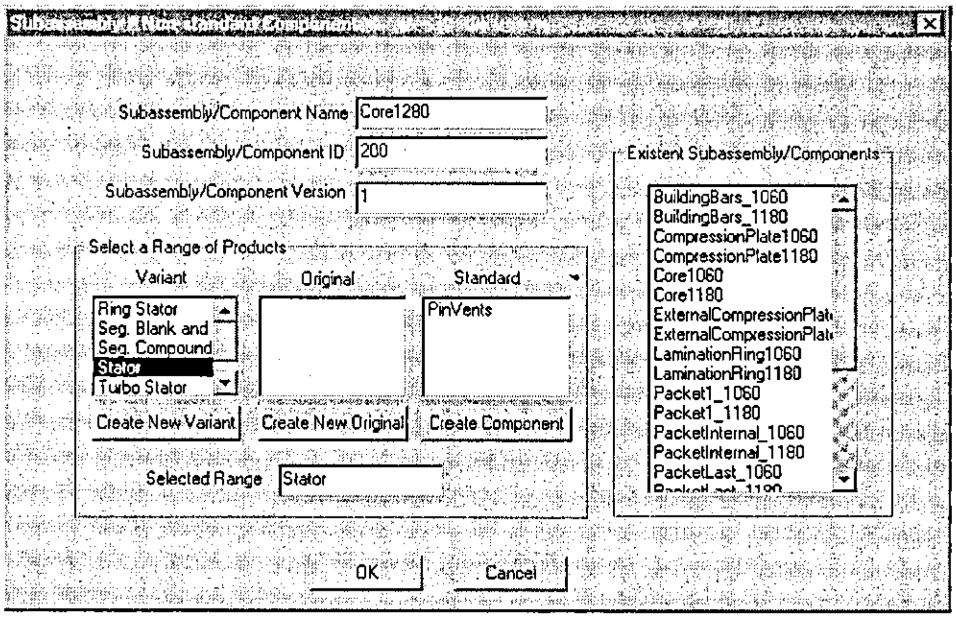



Figure 9.7 Creation of a subassembly

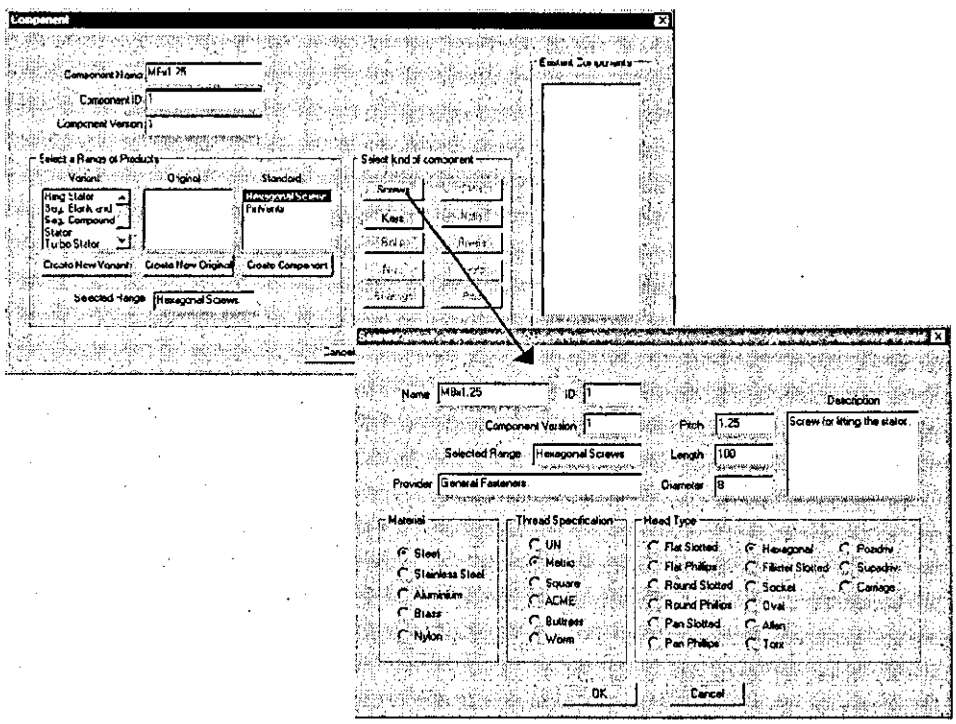



Figure 9.8 Creation of a Standard Component

Detailed information of the existent products can be accessed to provide the designer with information of similar products, components or subassemblies, which is useful for reusing information. Figure 9.9 shows how the designer can define the Geometry for the new *Stator1280* based on information existent in the Product Model for the *Stator1180*.

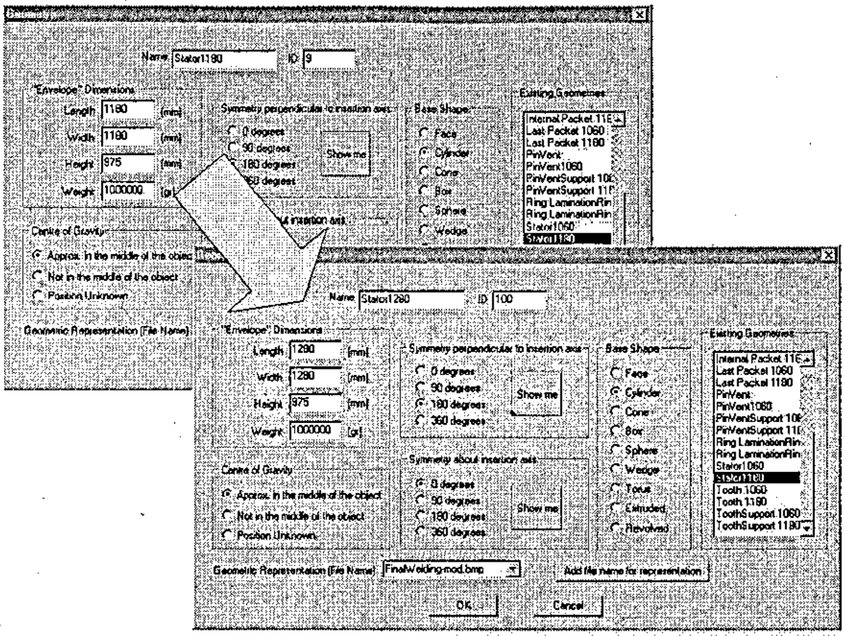



Figure 9.9 Reuse of information existent in the Product Model

Figure 9.9 showed the value of having the products classified in Ranges to allow the reuse of information stored in the Product Model. The information that can commonly be reused is the one that defines a product, grouped in the structure of the Product Model proposed in this research under the parent class Characteristics. The Characteristics class of the product was explained in section 7.2.1, and their structure in the Product Model is shown in Figure 9.10.

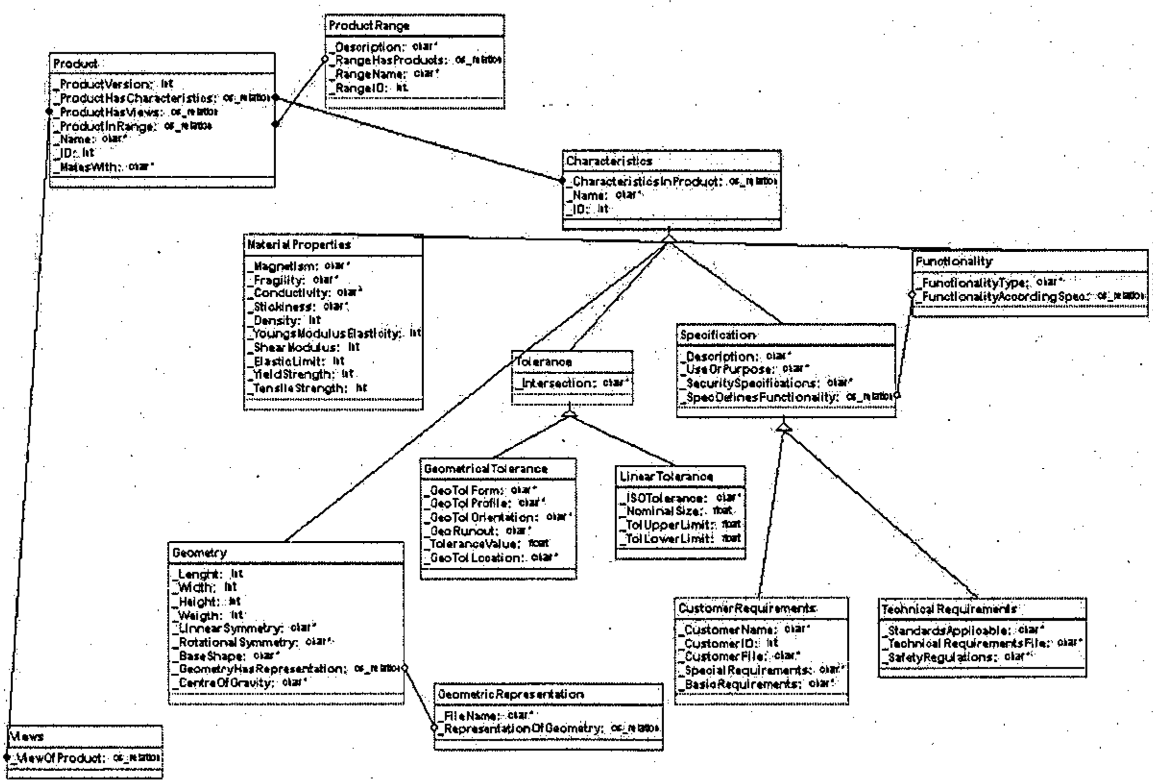



Figure 9.10 Detail of the Product Model Structure showing Characteristics

The Characteristics information for a product can be introduced as soon as the information is available, and it can be modified according to the progress in different stages of the design cycle. The dialog box that shows the Characteristics defined for a product is shown in different population stages in Figure 9.11.

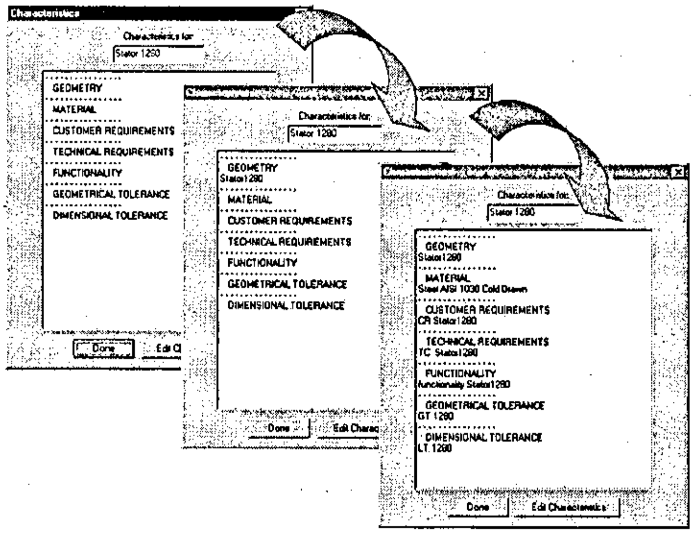



Figure 9.11 Progressive population of Characteristics

The Characteristics information defined for the product can be accessed at any time in order to edit, show, complete or update it, according to the stage of the product design, as depicted in Figure 9.12.

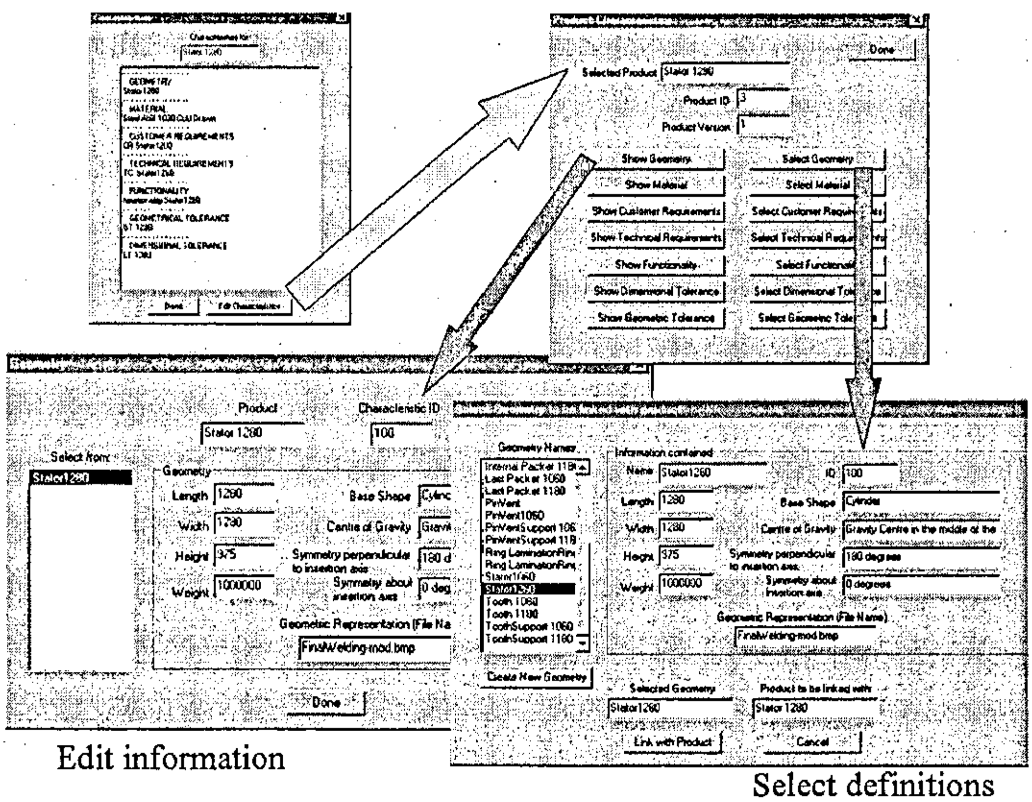



Figure 9.12 Editing and adding Characteristic information at all the stages of the Product Design cycle

### 9.3.2. Populating assemblies and subassemblies

The subassemblies and components can be linked to a parent assembly even if the information about them is very limited because, as mentioned in the previous section, this information can be added as it becomes available.

Section 9.2 explained the basic structure of the stators used for this test, in this section the connection between the *Stator1280* and the *Packet1_1280* is explored.

Figure 9.13 shows the dialog box in which the parent assemblies and subassemblies can be linked with their *children* subassemblies and components. In the example shown the 'Stator 1280' parent assembly is being linked with the subassembly 'Packet1_1280'.

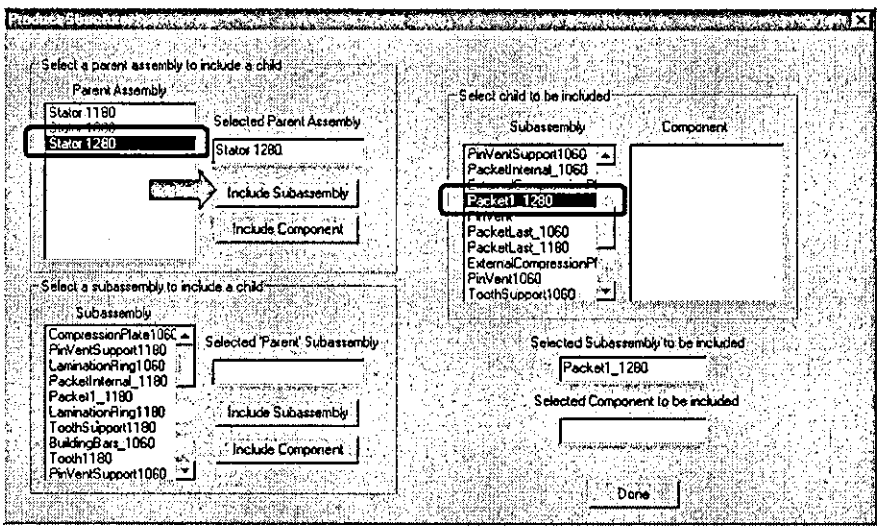



Figure 9.13 Include Subassembly in Parent Assembly

The information about the connection of the parts is stored in the Product Model in the Connecting class of the Design View. The connection between the parts can be either direct or indirect; if the latter occurs, it can be welded or threaded. The detailed structure for the Design Views in the Product Model is depicted in Figure 9.14.

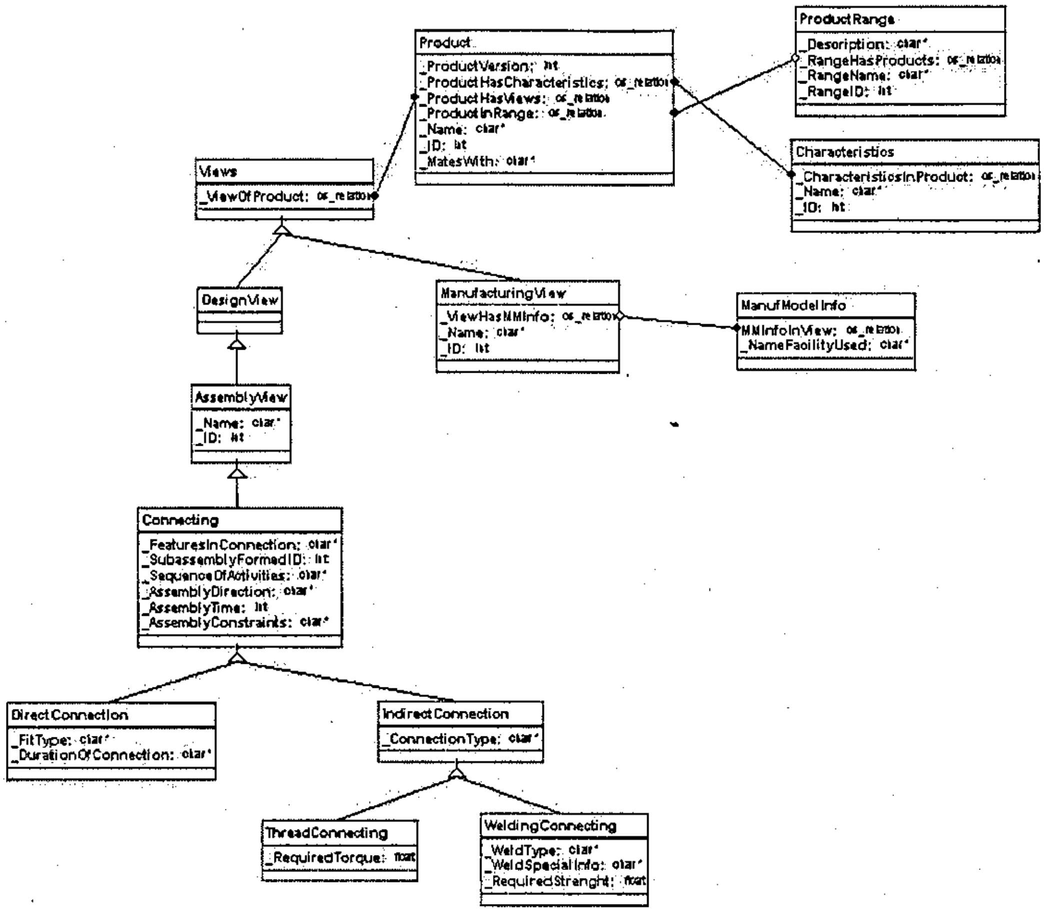



Figure 9.14 Detail of the Product Model Structure showing the Design View

For linking the *Stator1280* with the *Packet1_1280*, a direct connection was selected with the parameters shown in Figure 9.15.

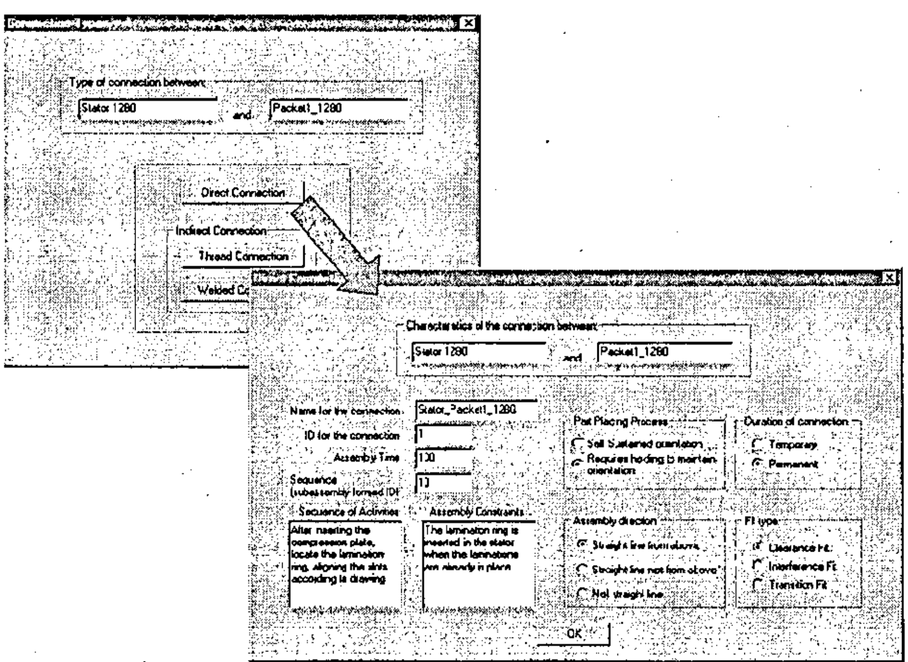



Figure 9.15 Creation of a connection between parts

When the connection has been created, the subassembly is linked to the parent assembly, and the sequence number indicated is used to show the order in which it has to be assembled. Figure 9.16 shows an assembly tree with the relation between the parts, it also shows the sequence order in which has to be assembled.

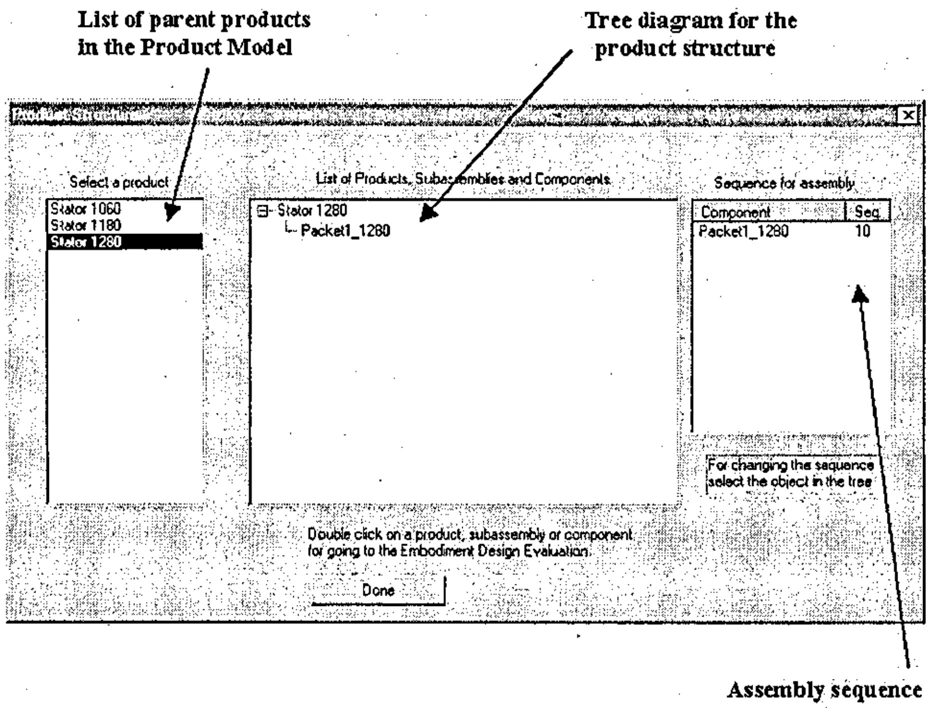



Figure 9.16 Product structure

The complete structure for the Stator1280 is depicted in Figure 9.17, showing the capability of holding the complete structure of a product in the Product Model. All the information of the product can be edited and completed at any stage of the product development.

### 9.3.3. Case study results

This test has shown that the structure created for the Product Model is able to hold the information necessary to define a product, including its characteristics and the connections that exist between parts, assemblies and subassemblies. This information is sufficient for defining the product in terms of its assembly.

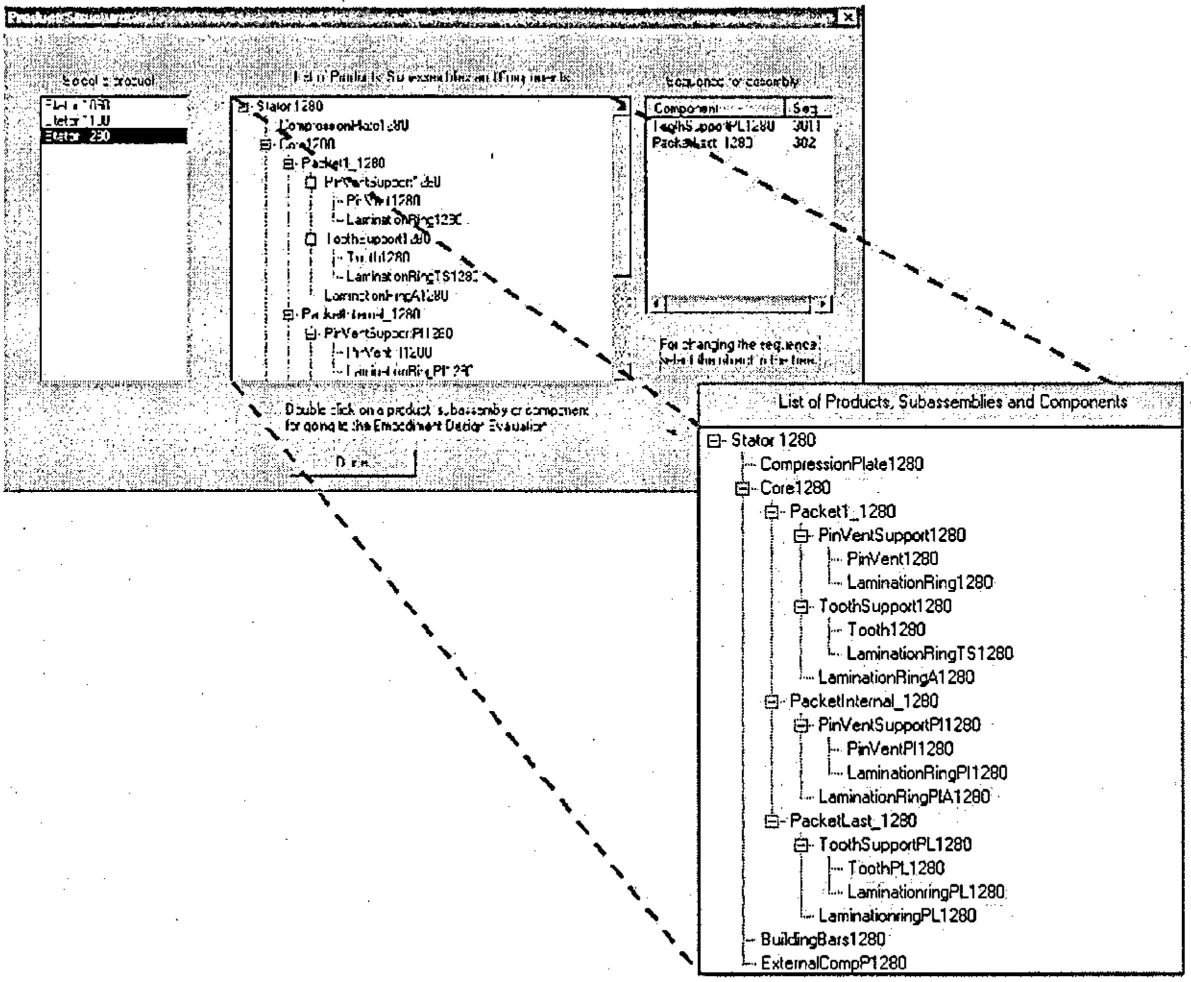



Figure 9.17 Structure for the Stator 1280

The classification of the products in Ranges allows the creation of product families that can help the designer in the process of the creation of new or variant products.

The use of the structure defined for the Product Model allows the progressive population of the product's characteristics, this is necessary for a concurrent environment in which the information has to be available to other areas as soon as it has been defined. The progressive population also allows the system to provide recommendations from the early stages of product design. The assembly recommendations are explored in section 9.6.

## 9.4. Manufacturing Model Structure for Supporting Assembly

This section reports a test aimed at the following

- To evaluate to what extent the assembly facilities of an enterprise can be captured in the levels of the Manufacturing Model.
- To describe how the Resources, Processes and Strategies can be defined and instantiated at the station level of the facility structure.
- To explore the relations between Resources, Processes and Strategies within the Manufacturing Model.

### 9.4.1. Populating different levels of the facility structure

As explained in section 7.3, the facility levels defined for the Manufacturing Model are Enterprise, Factory, Shop, Cell, and Station. Figure 9.18 shows the relations between those facility levels and the relations that have with the Resources, Processes, and Strategies used for defining them.

The first stage of this test consists on the creation of different levels of facilities, in order to show the aggregation relationships defined for them in the structure of the Manufacturing Model, and how the *parent* and *children* facilities are selected.

In order to populate the Manufacturing Model with the facilities in different levels, it is necessary to start defining the top level, which will contain other facility levels. The top level defined for the facilities is the Enterprise; its inclusion in the Manufacturing Model is shown in Figure 9.19.

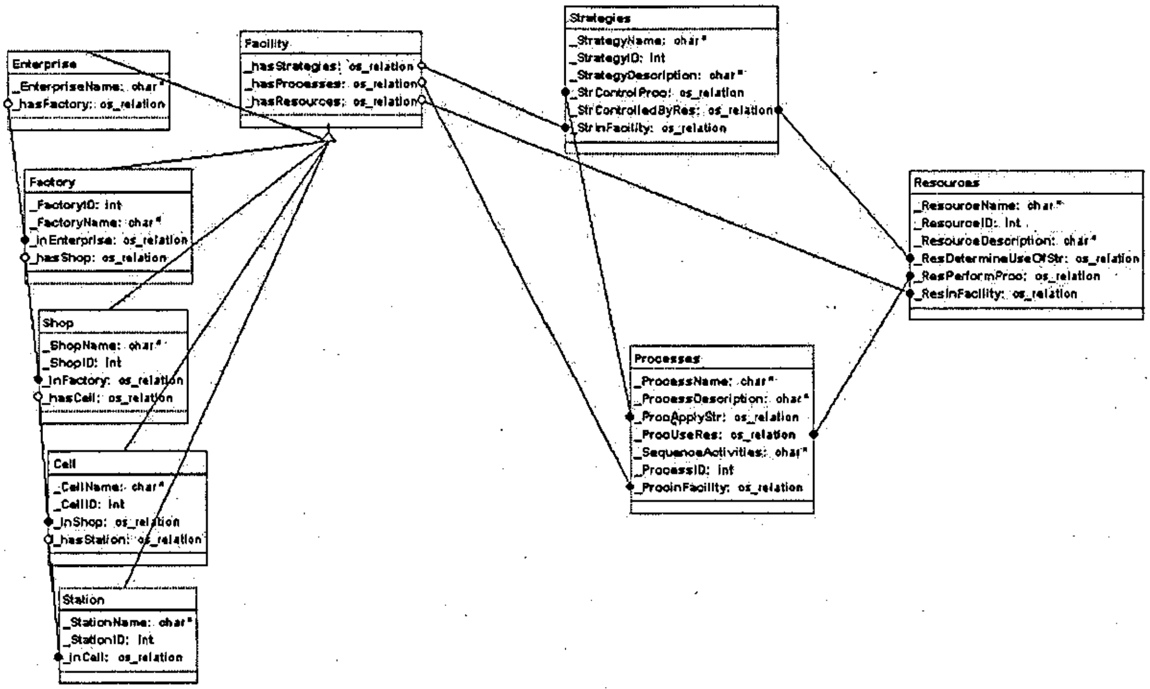



Figure 9.18 Detail of the Manufacturing Model Structure showing Facility Levels

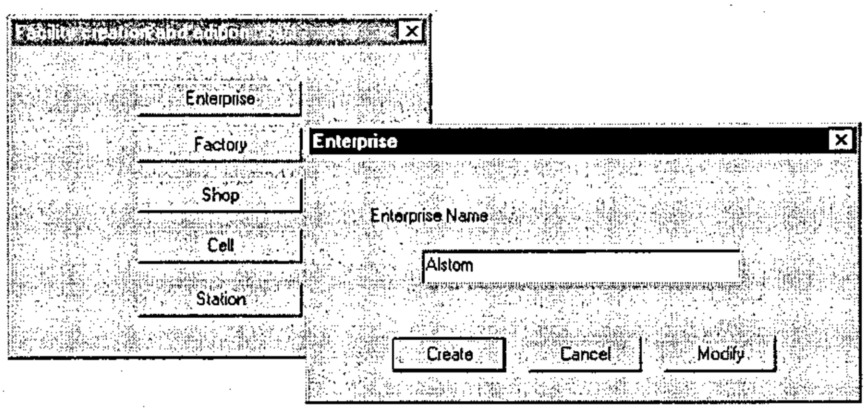



Figure 9.19 Definition of Enterprise

The creation of other levels has to be done in a progressive sequence, starting with Factories and finishing with Cells. When the facilities are instantiated, the name and ID are introduced to the Manufacturing Model. The previous levels of the structure are selected from a list that shows the information obtained from the Manufacturing Model. The populating process is shown in Figure 9.20.

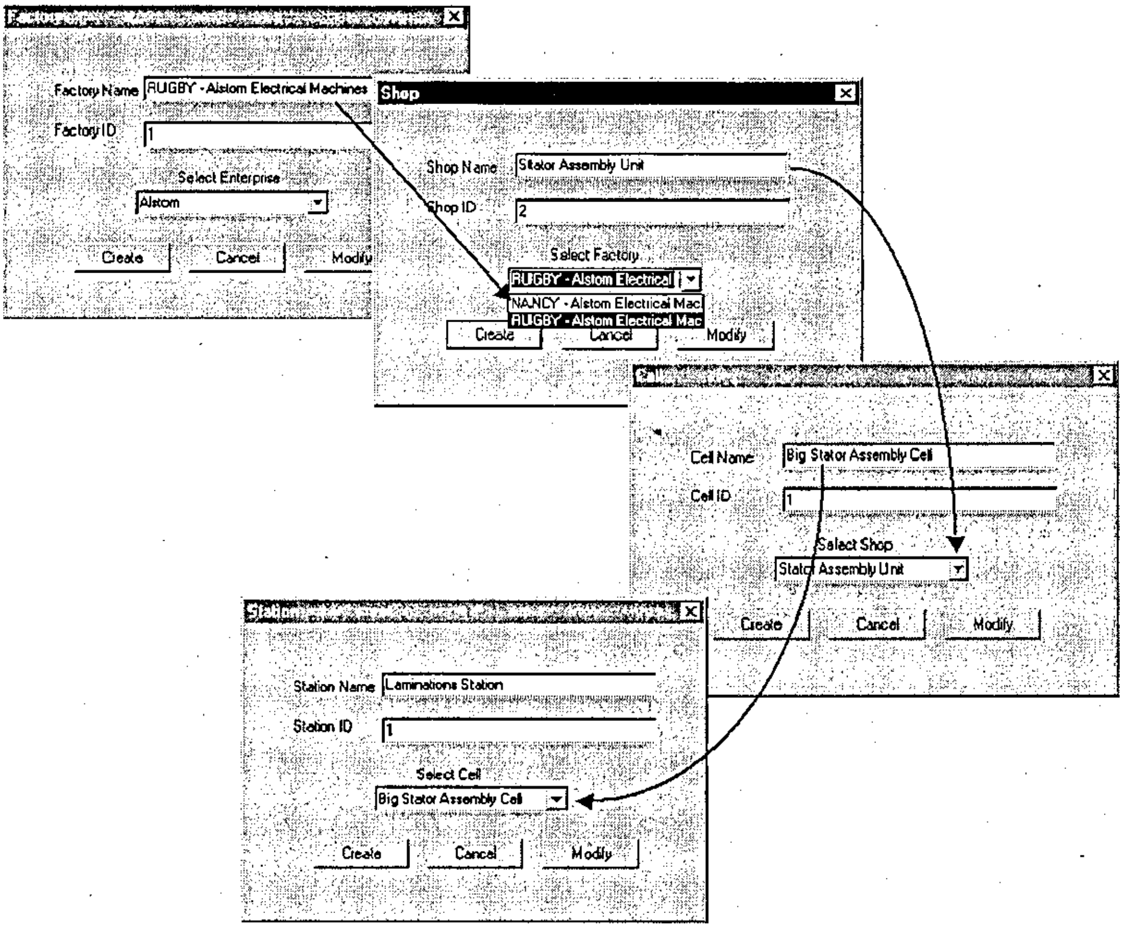



Figure 9.20 Creation and linking of facilities at different levels

The result of the population process is shown in Figure 9.21 using the ObjectStore Inspector.

| | EnterpriseName | FactoryName | ShopName | CellName | StationName |
|---|---|---|---|---|---|
| 1 | Alstom | RUGBY - Alstom Electrical Machines | Assembly Unit | Stator Assembly | Medium Stators |
| 2 | | | | | Big Stators |
| 3 | | | | Rotor Assembly | |
| 4 | | | Stator Assembly Unit | Big Stator Assembly Cell | Laminations Station |
| 5 | | | Rotor Assembly Unit | | |
| 6 | | NANCY - Alstom Electrical Machines | | | |

Figure 9.21 Facilities information in the Manufacturing Model

Resources, Processes, and Strategies can be captured at different levels of facility, this allows the Manufacturing Model to support activities at different levels of the organisation. The population of Resources, Processes and Strategies is explained in the following sections.

### 9.4.2. Populating Resources

Resources are the physical means used for performing the operations. The assembly resources included in the structure for the Manufacturing Model include Transporting, Storage, and Assembly Equipment. For the testing software the Assembly Equipment included in the structure is related directly to the assembly of stators for electric generators, such as the Mandrel and Press. The Manufacturing Model structure for the Resources is depicted in Figure 9.22.

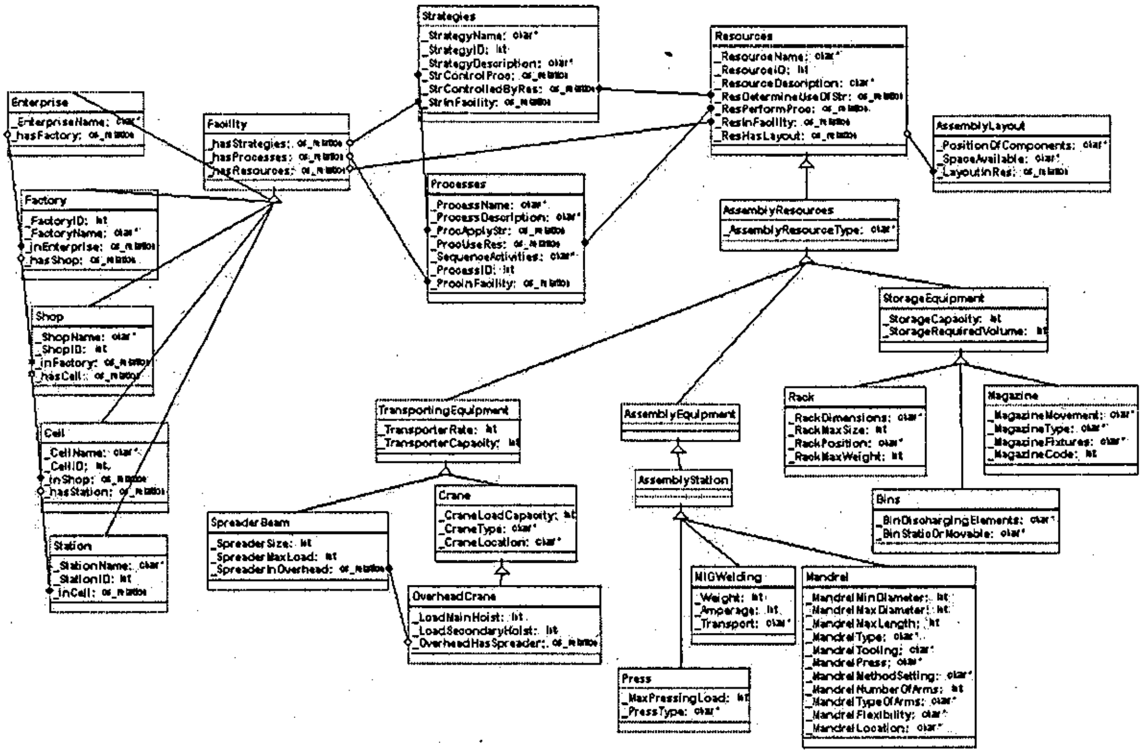
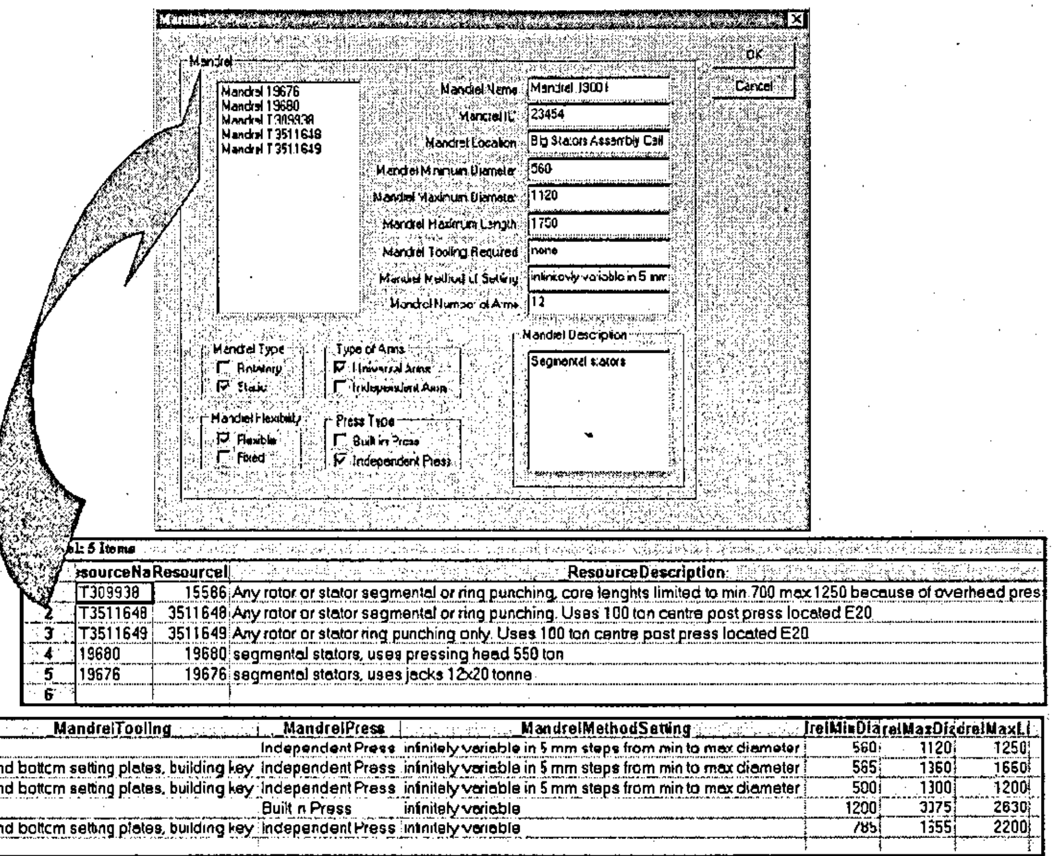



Figure 9.22 Detail of the Manufacturing Model Structure showing Resources

Resources are populated into the Manufacturing Model by using dialog boxes similar to the examples shown in Figure 9.23 and Figure 9.24. The first shows the creation of a Mandrel, this apparatus is used for aligning the laminations that conform the packets, mentioned in section 9.2. The second illustrates the inclusion of a crane information in the Manufacturing Model.

With the inclusion of Mandrel information into the Manufacturing Model, it is possible for the user to access the information relating to other mandrels defined previously. Figure 9.23 shows the dialog box in which the user introduces the information and, in the lower part of the figure, the information already contained in the Manufacturing Model read with the Objectstore Inspector.

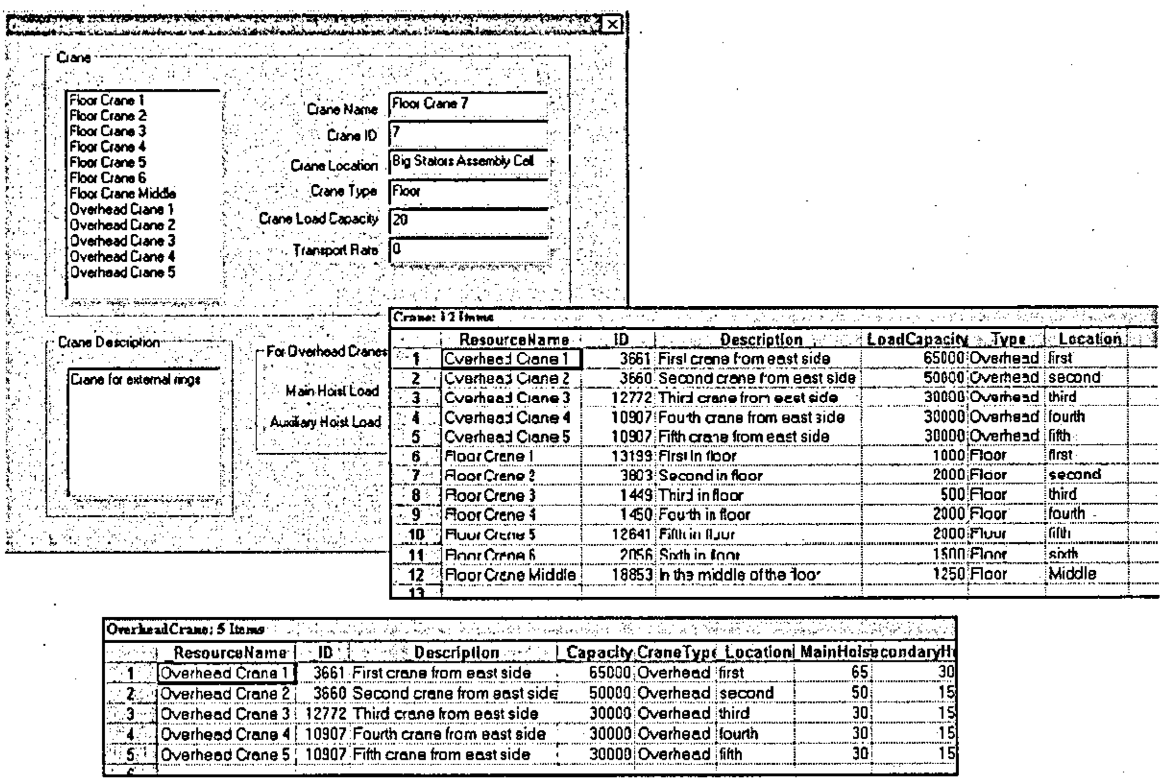



l: 5 Items

| | sourceNa | Resourcel | ResourceDescription |
|---|---|---|---|
| 1 | T309938 | 15586 | Any rotor or stator segmental or ring punching, core lenghts limited to min 700 max 1250 because of overhead pres |
| 2 | T3511648 | 3511648 | Any rotor or stator segmental or ring punching. Uses 100 ton centre post press located E20 |
| 3 | T3511649 | 3511649 | Any rotor or stator ring punching only. Uses 100 ton centre post press located E20 |
| 4 | 19680 | 19680 | segmental stators, uses pressing head 550 ton |
| 5 | 19676 | 19676 | segmental stators, uses jacks 12x20 tonne |
| 6 | | | |

| MandrelTooling | MandrelPress | MandrelMethodSetting | relMinDia | relMaxDia | drelMaxL |
|---|---|---|---|---|---|
| none | Independent Press | infinitely variable in 5 mm steps from min to max diameter | 560 | 1120 | 1250 |
| top and bottcm setting plates, building key | Independent Press | infinitely variable in 5 mm steps from min to max diameter | 565 | 1360 | 1660 |
| top and bottcm setting plates, building key | Independent Press | infinitely variable in 5 mm steps from min to max diameter | 500 | 1300 | 1200 |
| none | Built n Press | infinitely variable | 1200 | 3375 | 2630 |
| top and bottcm setting plates, building key | Independent Press | infinitely variable | 785 | 1355 | 2200 |

Figure 9.23 Creation of Assembly Resources: Mandrel




Crane: 12 Items

| | ResourceName | ID | Description | LoadCapacity | Type | Location |
|---|---|---|---|---|---|---|
| 1 | Cverhead Crane 1 | 3661 | First crane from east side | 65000 | Overhead | first |
| 2 | Cverhead Crane 2 | 3660 | Second crane from east side | 50000 | Overhead | second |
| 3 | Cverhead Crane 3 | 12772 | Third crane from east side | 30000 | Overhead | third |
| 4 | Cverhead Crane 4 | 10907 | Fouth crane from east side | 30000 | Overhead | fourth |
| 5 | Cverhead Crane 5 | 10907 | Fifth crane from east side | 30000 | Overhead | fifth |
| 6 | Floor Crene 1 | 13199 | First in floor | 1000 | Floor | first |
| 7 | Floor Crene 2 | 3803 | Second in floor | 2000 | Floor | second |
| 8 | Floor Crene 3 | 1449 | Third in floor | 500 | Floor | third |
| 9 | Floor Crene 4 | 1450 | Fouth in floor | 2000 | Floor | fourth |
| 10 | Fluur Crene 5 | 12641 | Fifth in fluur | 2000 | Fluur | fifth |
| 11 | Floor Crene 6 | 2056 | Sixth in floor | 1500 | Floor | sixth |
| 12 | Floor Crene Middle | 18853 | In the middle of the floor | 1250 | Floor | Middle |
| 13 | | | | | | |

OverheadCrane: 5 Items

| | ResourceName | ID | Description | Capacity | CraneType | Location | MainHois | econdaryH |
|---|---|---|---|---|---|---|---|---|
| 1 | Overhead Crane 1 | 3661 | First crane from east side | 65000 | Overhead | first | 65 | 30 |
| 2 | Overhead Crane 2 | 3660 | Second crane from east side | 50000 | Overhead | second | 50 | 15 |
| 3 | Overhead Crane 3 | 12772 | Third crane from east side | 30000 | Overhead | third | 30 | 15 |
| 4 | Overhead Crane 4 | 10907 | Fourth crane from east side | 30000 | Overhead | fourth | 30 | 15 |
| 5 | Overhead Crane 5 | 10907 | Fifth crane from east side | 30000 | Overhead | fifth | 30 | 15 |
| 6 | | | | | | | | |

Figure 9.24 Creation of Assembly Resources: Crane

### 9.4.3. Populating Processes

The assembly Processes structure of the Manufacturing Model includes Handling and Fitting Processes, as shown in Figure 9.25. The fitting processes include Compose and Fastening processes, such as Welding and Threaded unions.

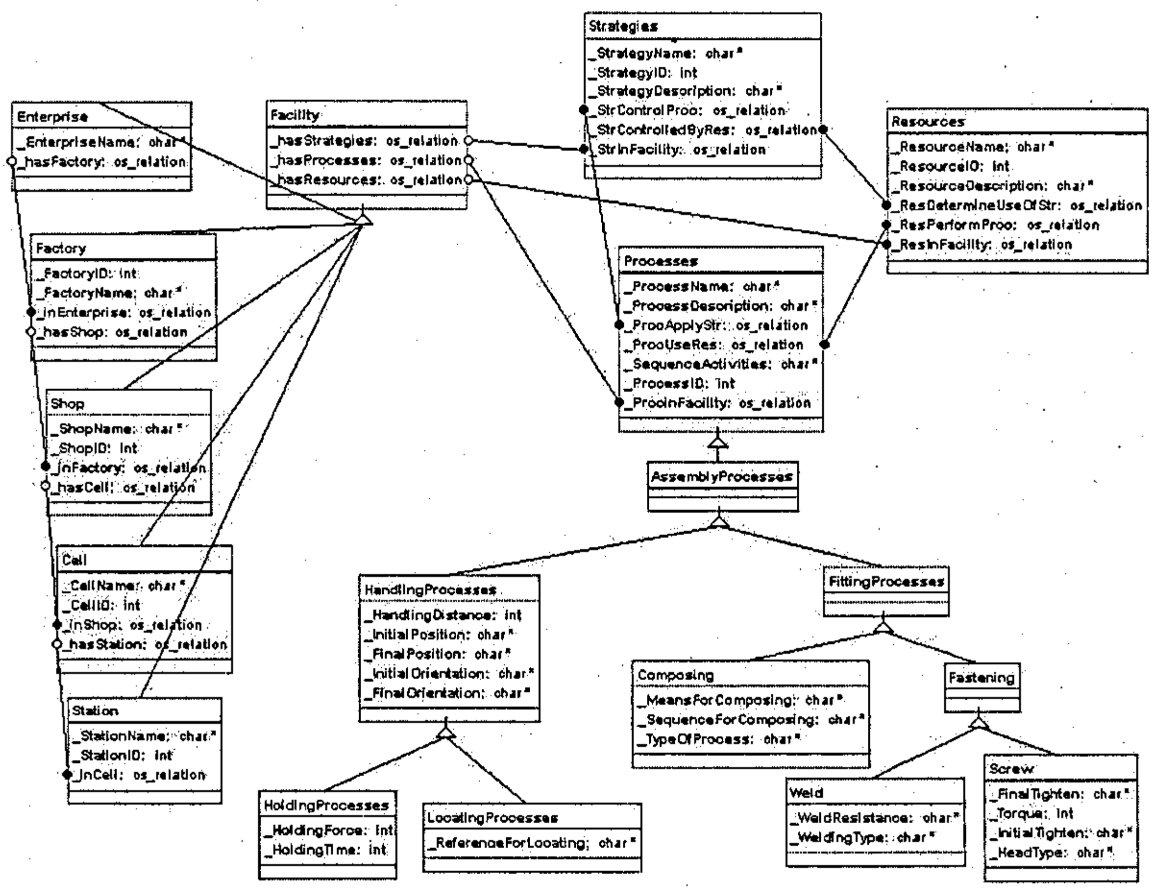
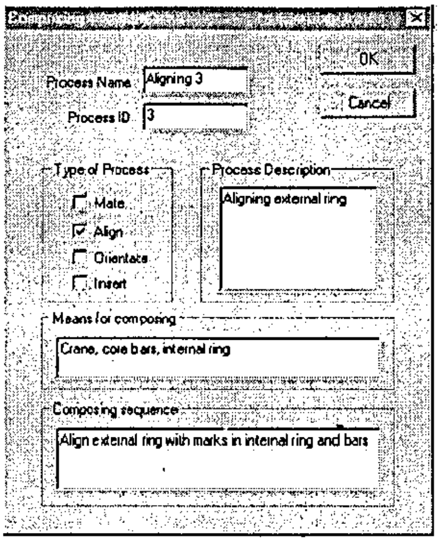



Figure 9.25 Detail of the Manufacturing Model Structure showing Processes

Figure 9.26 shows the population of a Composing Process and the information held by the Manufacturing Model.




Composing: 10 Items

| | ProcessName | ProcessDescription | Type | SequenceForComposing | MeansForComposing |
|---|---|---|---|---|---|
| 1 | composing 1 | description | Mate | sequence 1 | means composing 1 |
| 2 | composing 2 | description composing 2 | Align | sequence 2 | means composing 2 |
| 3 | Comp M 19676 | description composing 2 | Insert | Sequence for assembling the laminations, bars, etc. | Composing for mandrel 19676 |
| 4 | Comp M 19680 | description composing 3 | Insert | Sequence for assembling the laminations, bars, etc. | Composing for mandrel 19680 |
| 5 | Comp PH 550 | description pressing head 550 | Insert | Sequence for pressing the core laminations | Composing for pressing head 550 |
| 6 | Comp M T3511649 | description composing 3 | Insert | Sequence for assembling the laminations, bars, etc. | Composing for mandrel T3511649 |
| 7 | Comp M T3511648 | description composing 4 | Insert | Sequence for assembling the laminations, bars, etc. | Composing for mandrel T3511648 |
| 8 | Comp M T309938 | description composing 5 | Insert | Sequence for assembling the laminations, bars, etc. | Composing for mandrel T309938 |
| 9 | Comp PJacks 20 | description pressing jacks 20 | Insert | Sequence for pressing the core laminations | Composing for pressing jacks 20 |
| 10 | Aligning 3 | Aligning external ring | Align | Align external ring with marks in internal ring and bars | Crane, core bars, internal ring |

Figure 9.26 Creation of Assembly Processes: Composing

### 9.4.4. Populating Strategies

The structures in the Manufacturing Model define which Resources can be used for performing a particular Process, and which Processes can be performed with the existent Resources.

The Assembly Processes structure of the Manufacturing Model defined for the demonstration software includes Method of Transporting, Method of Insertion, and Method of Storage, as shown in Figure 9.27.

The strategies included in the Manufacturing Model structure belong to a Cell or Station level, except the Type of System strategy, which belongs to a higher level of the Facility structure. This strategy gives the advice on which kind of system has to be used, according to the characteristics of the product to be assembled. The interaction of Product and Manufacturing Models information is explored in the test shown in section 9.5.

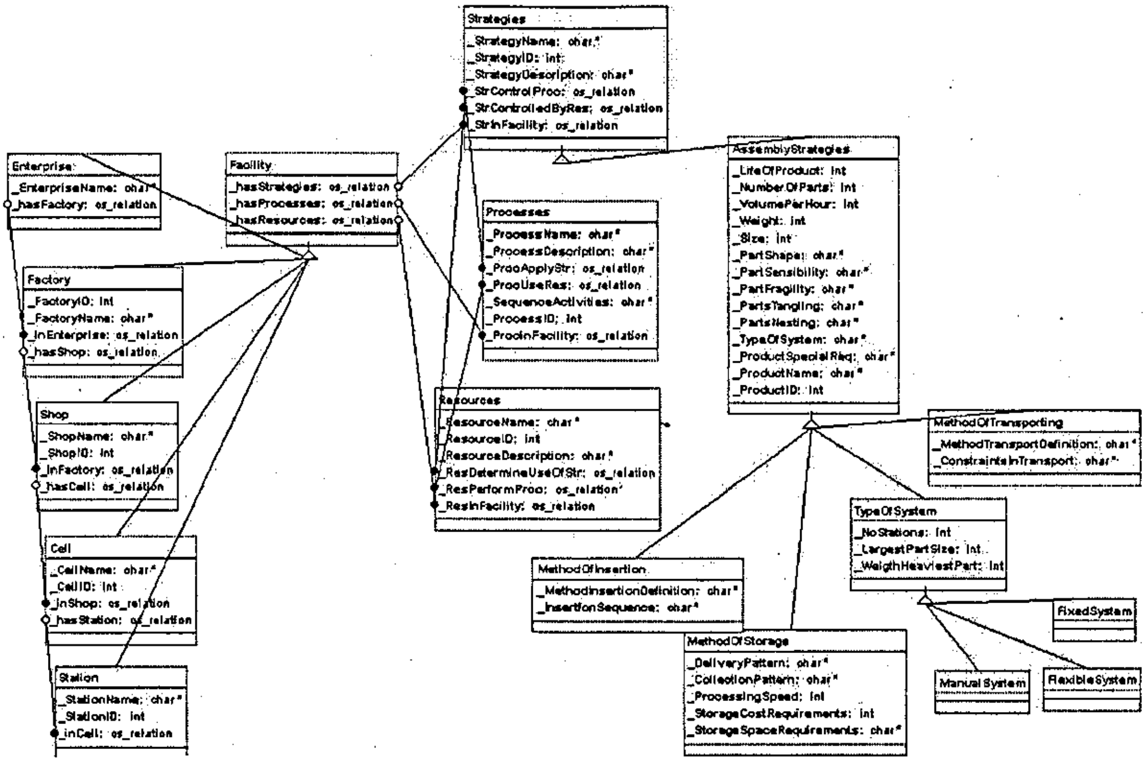



Figure 9.27 Detail of the Manufacturing Model Structure showing Strategies

The population of the Strategies defined for the demonstration software was done in a first stage as it is shown in Figure 9.28, in which the description, definition, and constraints for the strategy are given.

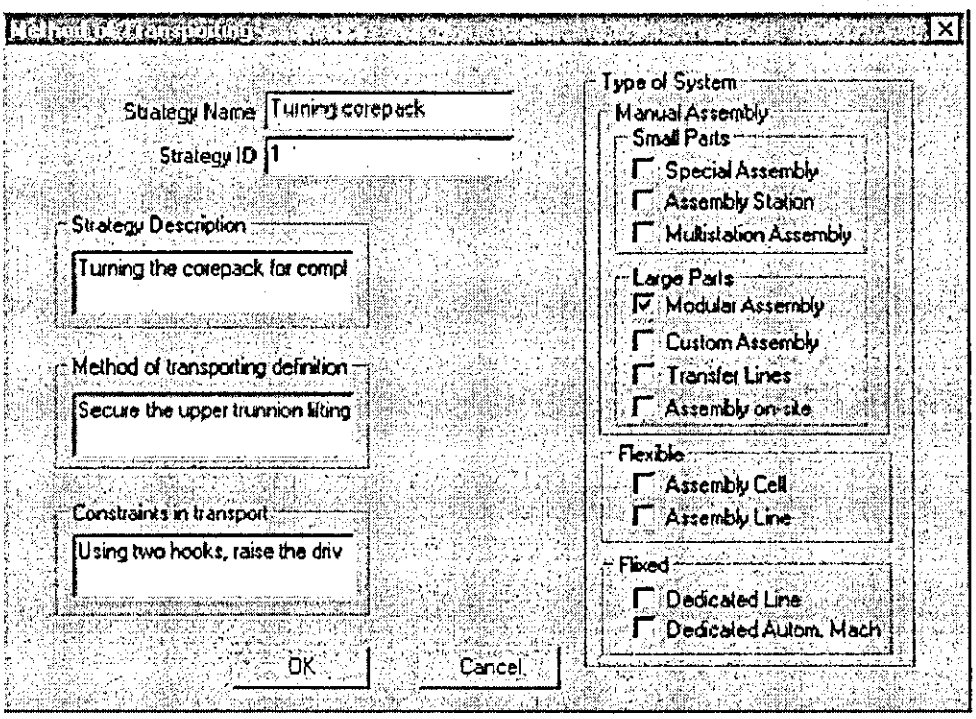



Figure 9.28 Creation of Assembly Strategies: Transporting

However, it was found to be necessary to have the product information available for every strategy definition. This proved the assumption made by the author that the assembly strategies are highly product-dependant, and in order to define them it is necessary to have information about the product for which the strategy is being defined. Section 9.5.5 explains how these product-dependant strategies are created.

### 9.4.5. Relations between Resources and Processes

The relations between Resources, Processes and Strategies defined in the structure of the Manufacturing Model explained in section 7.3 are shown in Figure 9.29.

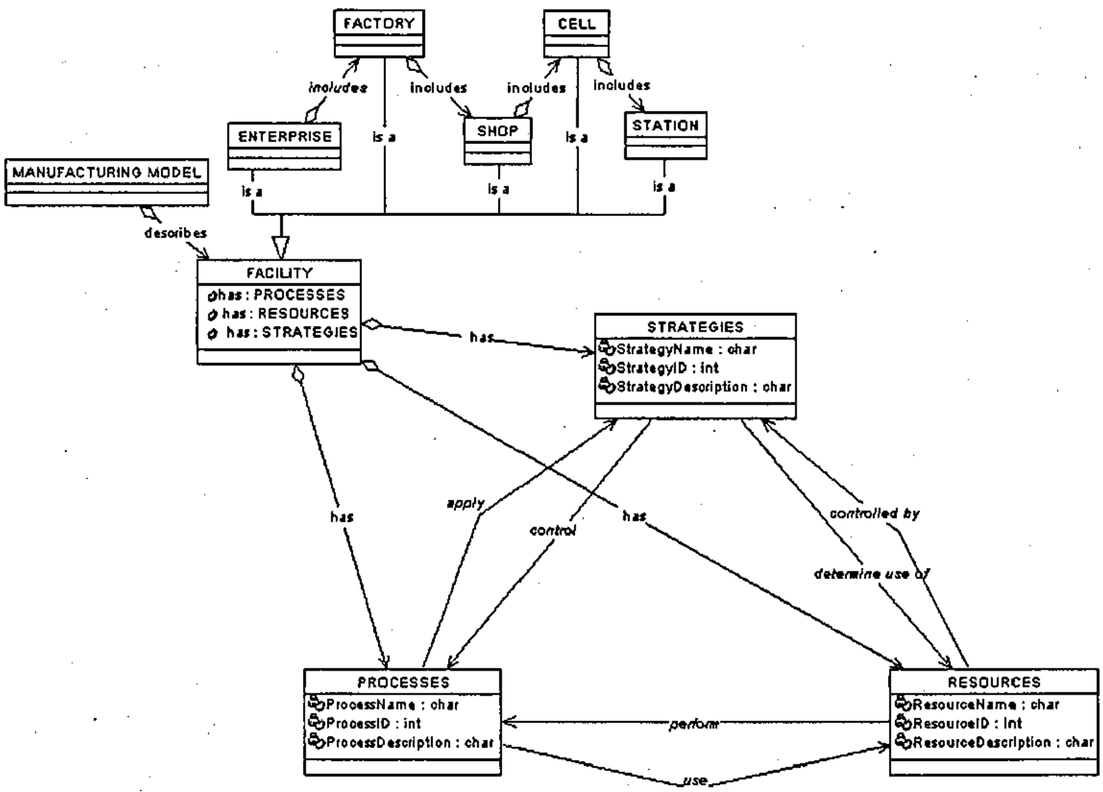



Figure 9.29 UML class diagram showing the relations between Resources, Processes and Strategies within the Facility

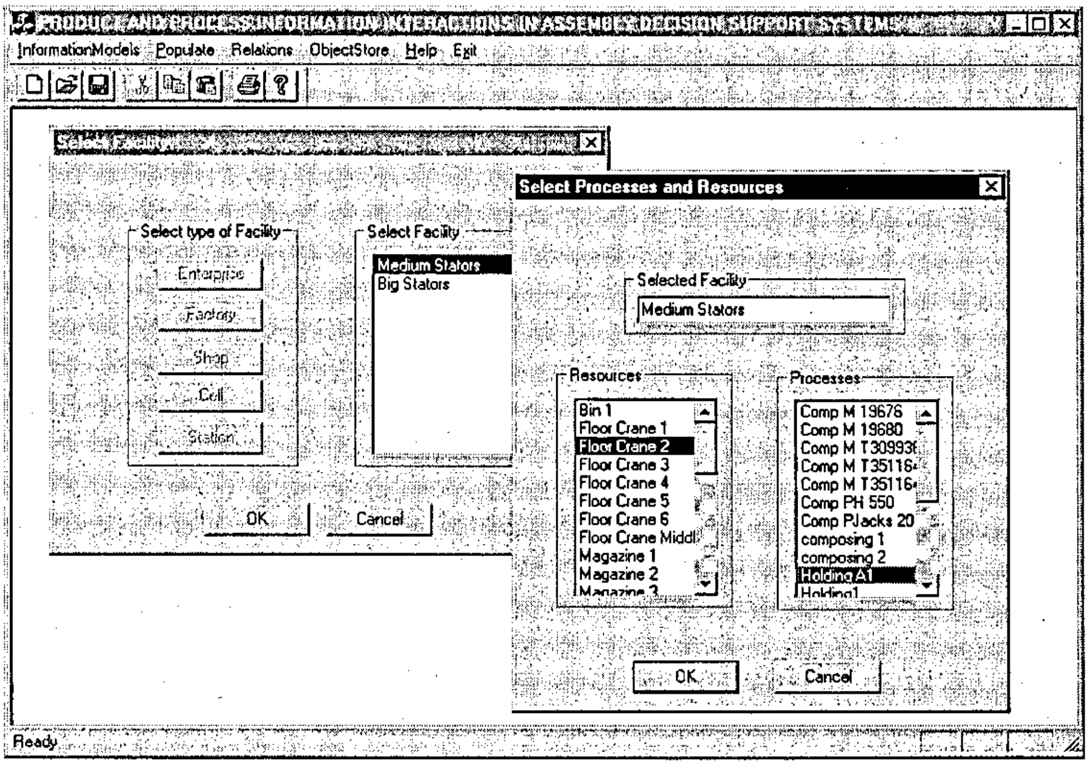
Those relations are captured in the Manufacturing Model by linking them with the Facilities, as shown in Figure 9.30.

### 9.4.6. Case Study Results

This test has shown that the data structure created for the Manufacturing Model can be populated with the information necessary to define the Assembly Resources, and Processes of a Facility.

This test showed that the Strategies information for Assembly is highly product-dependant, and it was necessary to modify the way in which the assembly strategies are defined. This is evaluated in Section 9.5.

It is possible to capture the relations between Resources, Processes, and Strategies in the Manufacturing Model with the Facilities to which they belong.




Station: 2 Items

| | StationName | ResourceNam | ProcessName |
|---|---|---|---|
| 1 | Medium Stators | Magazine 2 | Holding1 |
| 2 | | Magazine 1 | Holding A1 |
| 3 | | Floor Crane 2 | |
| 4 | Big Stators | Floor Crane 1 | Comp M T309938 |
| 5 | | | |

Figure 9.30 Relations between Processes and Resources in a Facility

## 9.5. Selecting Resources, Processes and Strategies for the Product

This section reports a test aimed at the following:

- To explain how the use of different viewpoints extend the capabilities of the Product Model
- To describe how the combined information structures of the Product Model and Manufacturing Model interact to support the selection of Resources, Processes, and Strategies for a product.
- To show how Strategies defined at different levels of the Facility can be used.

### 9.5.1. Relations between the Product Model and the Manufacturing Model

An important issue of this research was to extend the possibilities of interactions between the Product Model and the Manufacturing Model in order to allow an effective support of a Concurrent Engineering environment.

The use of Views in the Product Model has extended its capabilities by allowing a multiple viewpoint access to the information stored. Two Views were included in the demonstration software; the Design View was explained on section 7.2.3, and the way for populating it in section 9.3.2, and illustrated on Figure 9.14. The use of the Manufacturing View is explained in this section.

In order to establish a direct relationship between the Product Model and the Manufacturing Model, the Product Model has a set of classes that allow the storage of manufacturing information specific to the product, as was explained in section 7.4. In order to describe how this information is stored, the detail of the Manufacturing View class of the Product Model is depicted in Figure 9.31.

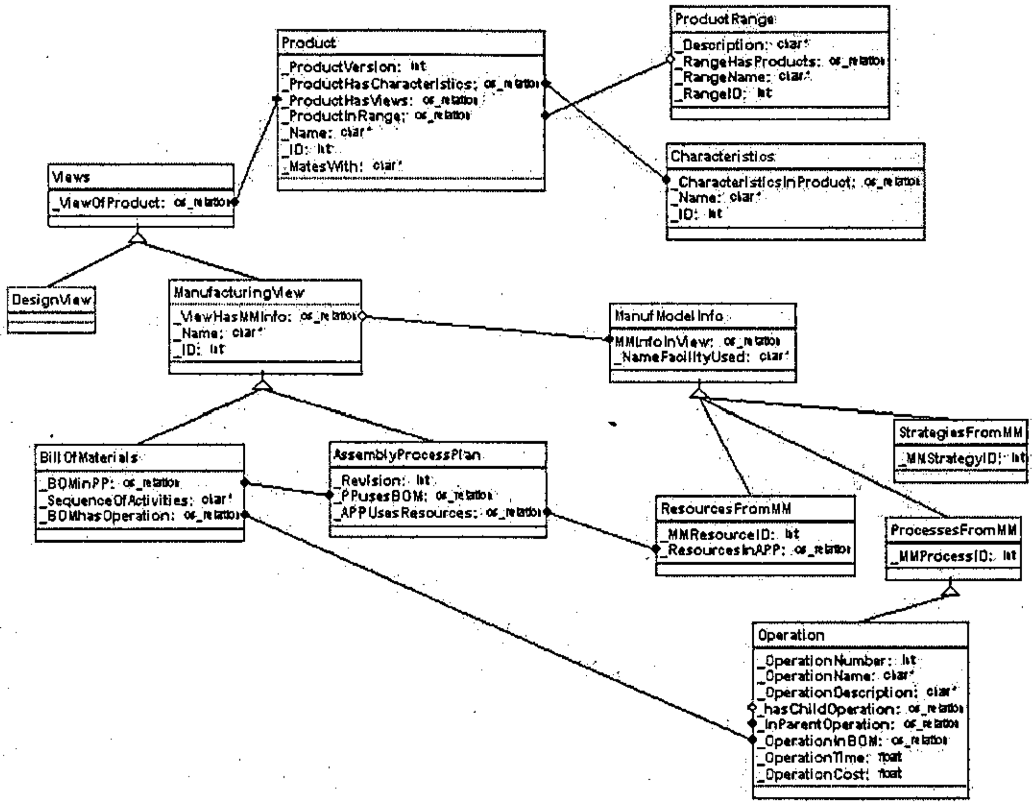



Figure 9.31 Detail of the Product Model Structure showing the Manufacturing View

The structure of the Manufacturing View of the Product Model for the testing system includes the Bill Of Materials and the Assembly Process Plan classes. The first stores the sequences of activities and the second the relation between the BOM and the Resources used in the assembly. The link between the Product Model and the Manufacturing Model is provided by the *ManufModelInfo* class, which has an association relationship with the *ManufacturingView* class. The children classes *ResourcesFromMM*, *StrategiesFromMM* and *ProcessesFromMM* have an *ID* attribute, which holds the relation with the Manufacturing Model. The author decided not to copy information from the Manufacturing Model into the Product Model, but to include an identifier, to reduce the maintenance required.

This test follows from the case study described in section 9.3. The test is divided in three sections, the first for the selection of Resources, the second for Processes and the last for Strategies. The selected information is stored in the Manufacturing View of the Product Model.

### 9.5.2. Access to Manufacturing Information

The user can access the information in the Manufacturing Model at any time in order to check the facilities available for the assembly of the product that is being designed, as shown in Figure 9.32. In fact, when the part is just being defined, the system is designed to ask the user if the information available should be displayed, this can be seen in Figure 9.41.

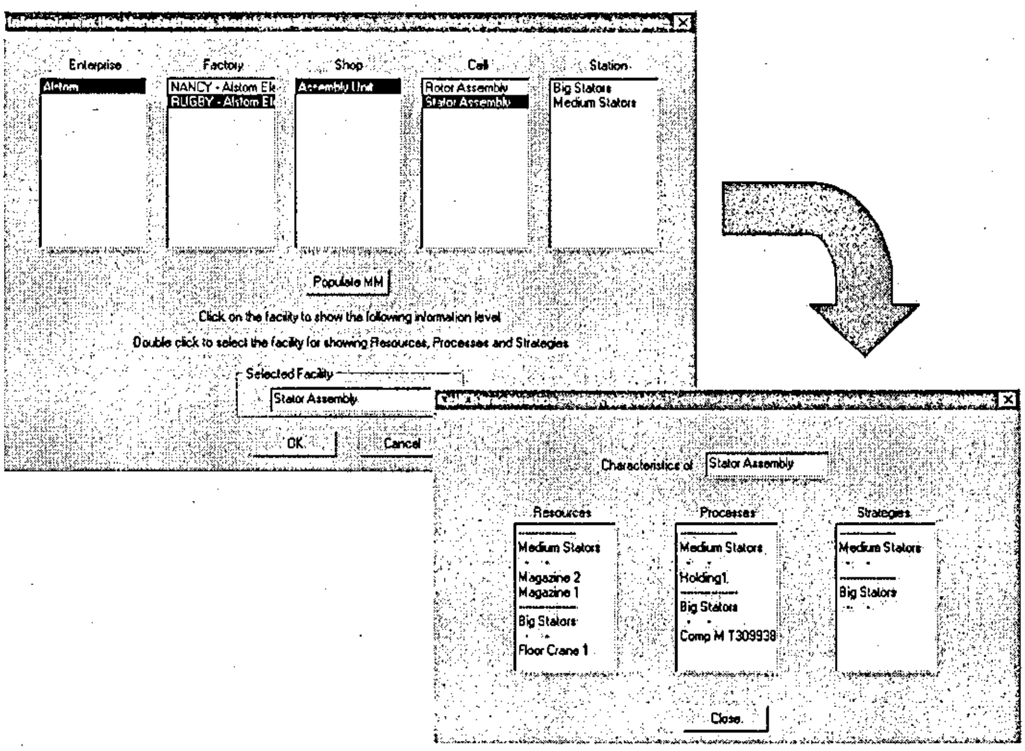



Figure 9.32 Access to the information existent in the Manufacturing Model

### 9.5.3. Selection of Assembly Resources

This aspect of the test shows how the Resources stored in the Manufacturing Model could be selected for a particular product.

In order to perform the selection of resources for a product, subassembly or component, it has to be chosen from the product structure tree diagram shown in Figure 9.33. The enterprise in which the assembly will be done is selected from the list. At this moment the Characteristics information of the Product Model for the selected part is used to analyse and filter the Resources existent in the Manufacturing Model, so the information shown as a result includes only Resources that comply with the characteristics of the part, as explained in section 7.4.3.

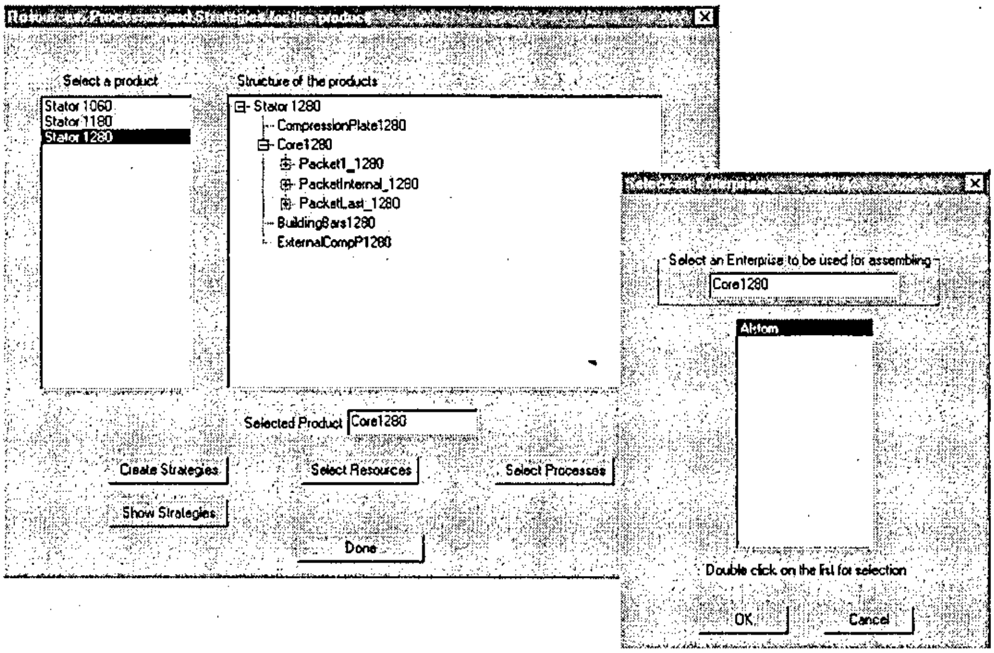


Figure 9.33 Selection of the enterprise to be used for assembling the product

The result of the filtering of information mentioned above is shown in Figure 9.34, from where the detail of the Resources information can be accessed before selecting it, an example is shown in Figure 9.35.

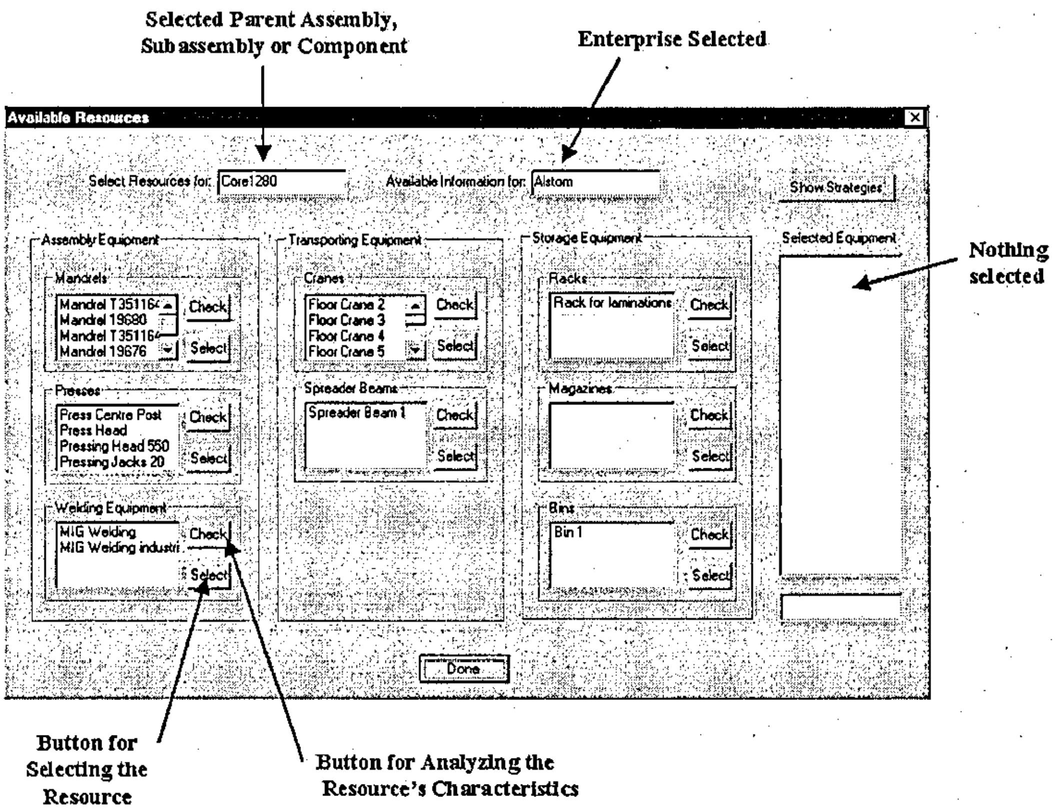



Figure 9.34 Selection of Assembly Resources

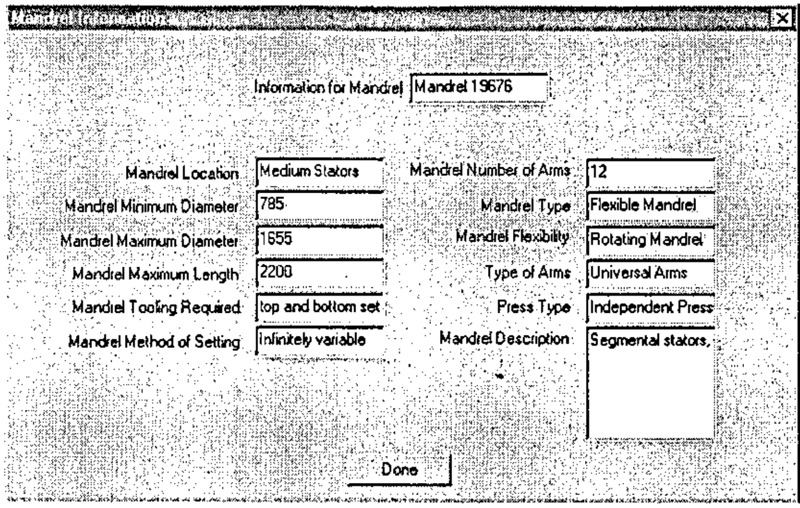



Figure 9.35 Example of Resources detailed information

As explained before, the Resources available for the product depend on the Characteristics of the part, in order to test the correct use of the Characteristics information, the geometry of the part analysed was modified, and the result of the change of Resources available is shown in Figure 9.36.

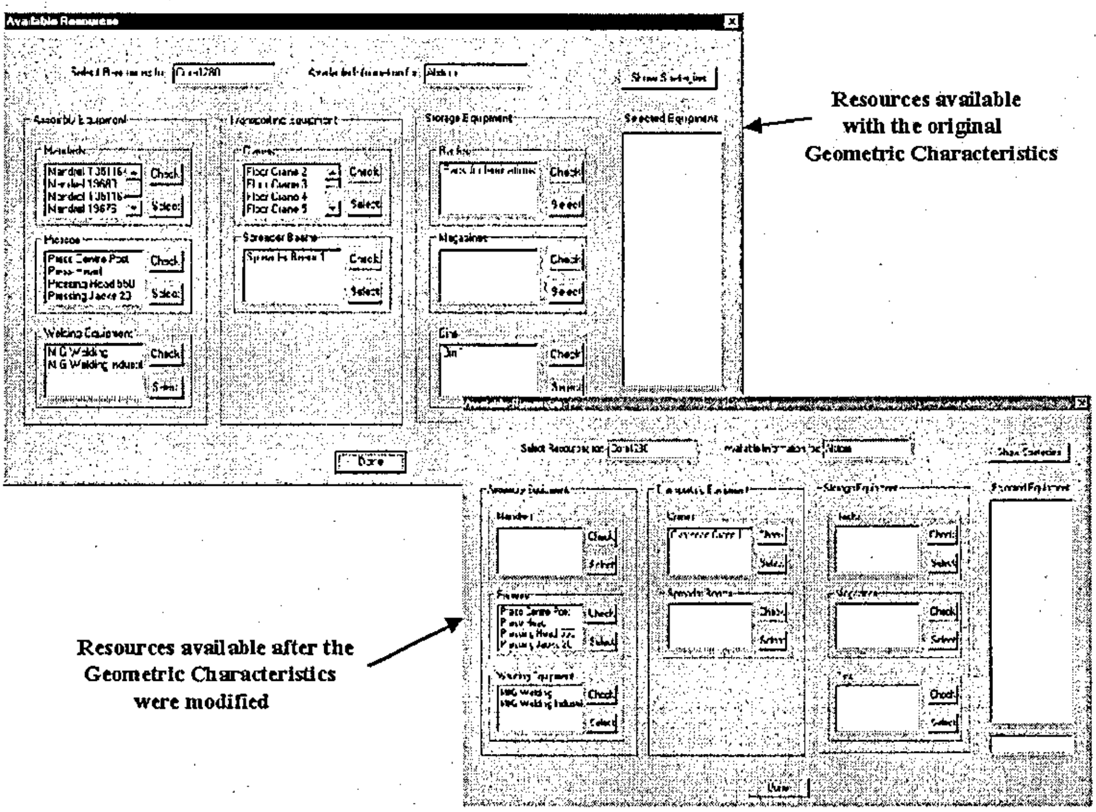



Figure 9.36 Change in the Resources available

The manufacturing information selected for the product is stored in the Manufacturing View of the Product Model, explained in section 7.4.3.

The selection of Resources can be done in different stages, according to the information available. The system provides the designer with information about the Resources already selected, which are shown in the dialog box. An example is shown in Figure 9.37.

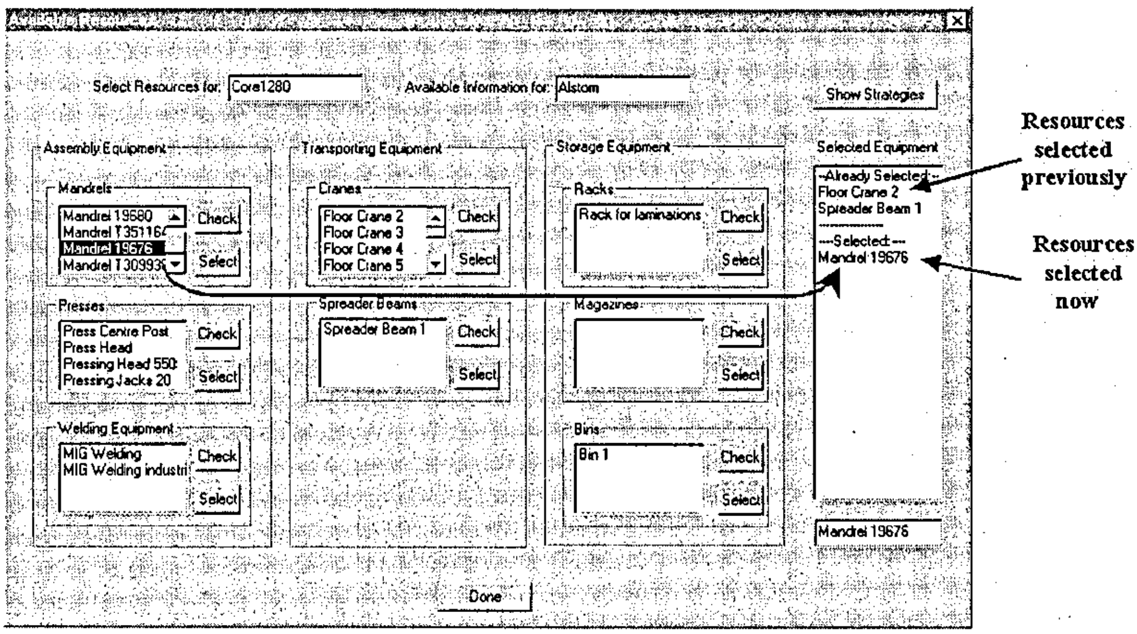



Figure 9.37 Selection of Resources in different stages

### 9.5.4. Selection of Assembly Processes

This test was performed to show how the Process information stored in the Manufacturing Model is used for supporting the design of a product.

The designer can access the information of the processes available in an enterprise at any moment of the design cycle, as shown in Figure 9.32. This allows the user to know which processes are already taking place, giving the possibility to design the product in such a way that similar processes could be applied.

The selection of particular processes for the product is similar to the selection of assembly resources, explained in section 9.5.5. In the selection of processes, the existent processes in the Manufacturing Model are filtered to show only the ones that comply with the characteristics of the part for which are selected, as it is shown in Figure 9.38.

It is also possible to check the processes information before selecting them for be applied in the product, by selecting the *'check'* button of the dialog box, as shown in Figure 9.39.

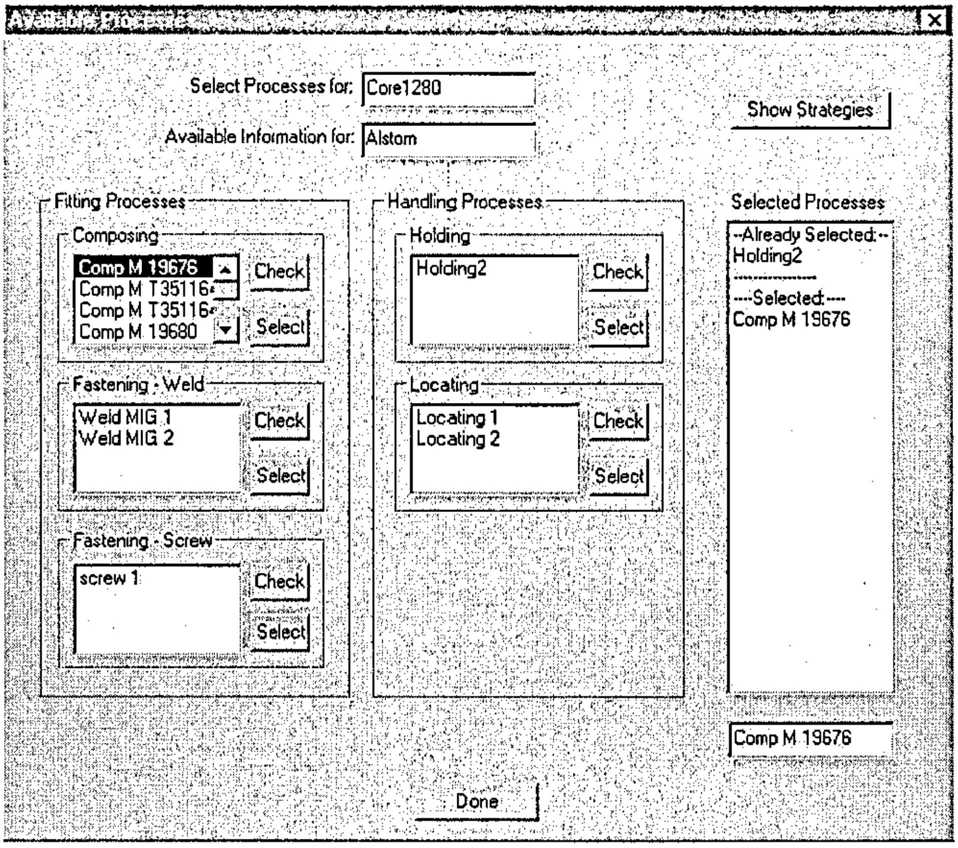



Figure 9.38 Selection of Assembly Processes

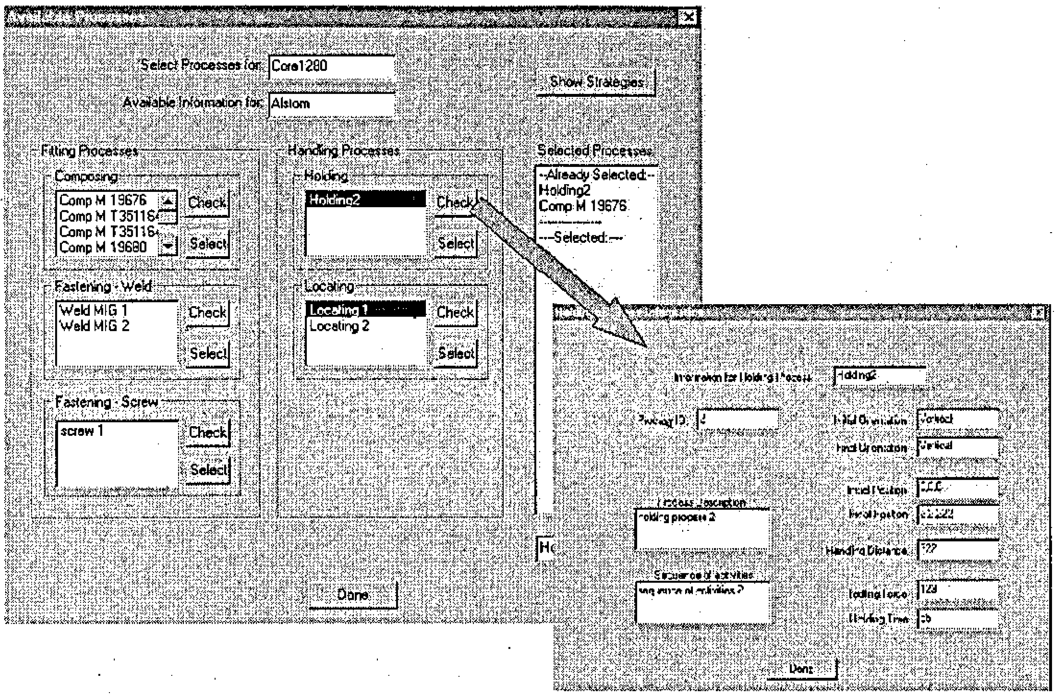


Figure 9.39 Check specifications for the Processes stored in the Manufacturing Model

### 9.5.5. Selection and Creation of Assembly Strategies

As mentioned in Section 9.4.4, the Type of System strategy belongs to a higher level of the Facility Structure. This strategy assists the designer from the beginning of the product design process by analysing the general characteristics of the product and matching them with the kind of system recommended for its assembly.

Once the name of the parent assembly has been introduced, and linked to a range of products, as shown in Figure 9.5 and Figure 9.6, the General Characteristics of the product have to be introduced as shown in Figure 9.40. This information is analysed by the system in order to recommend the kind of system to be used for its assembly, according to the Type of System Strategy information. The recommendation for the kind of system to be used in the assembly is shown in Figure 9.41.

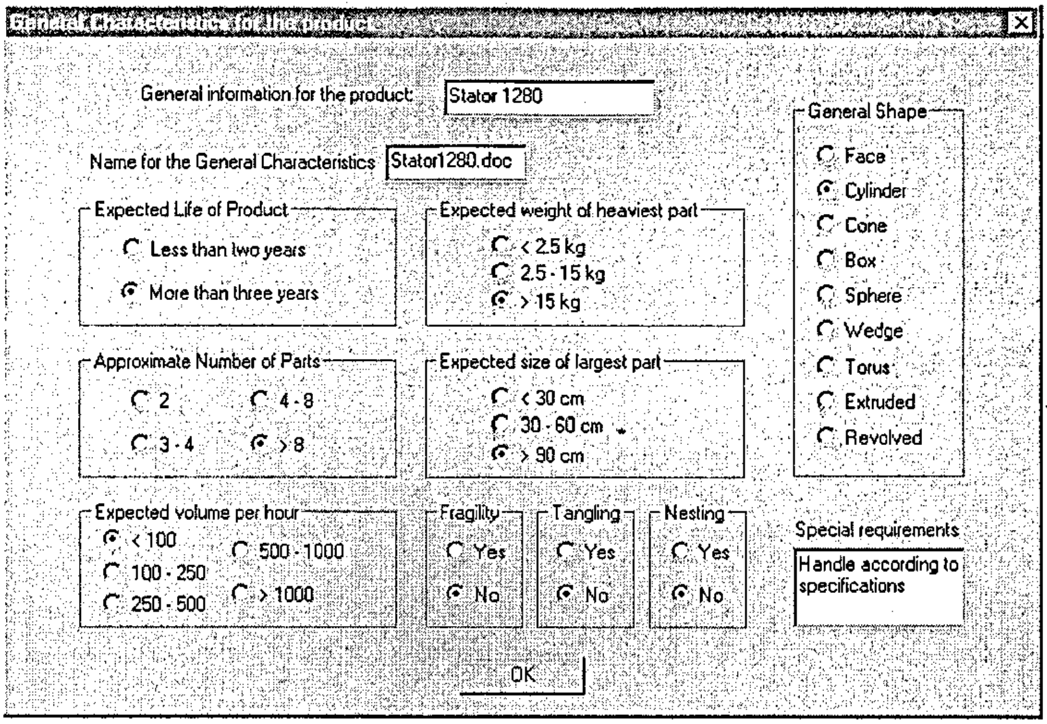



Figure 9.40 General characteristics for the product

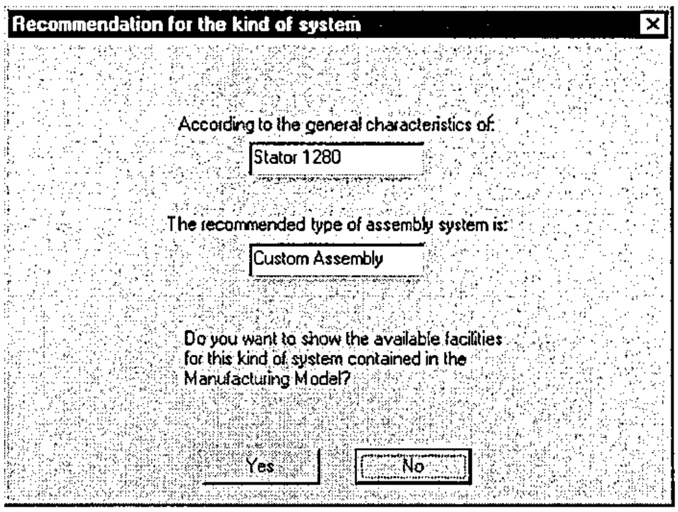



Figure 9.41 Recommendation of the type of assembly system

Section 9.4.4 explained that the definition of assembly Strategies requires knowing information about the product that will be assembled. In order to create the strategies for the products, the structure of the product is displayed on the screen of the testing system, so the user can select the parent assembly, subassembly or component for which the strategies will be created, as illustrated in Figure 9.42.

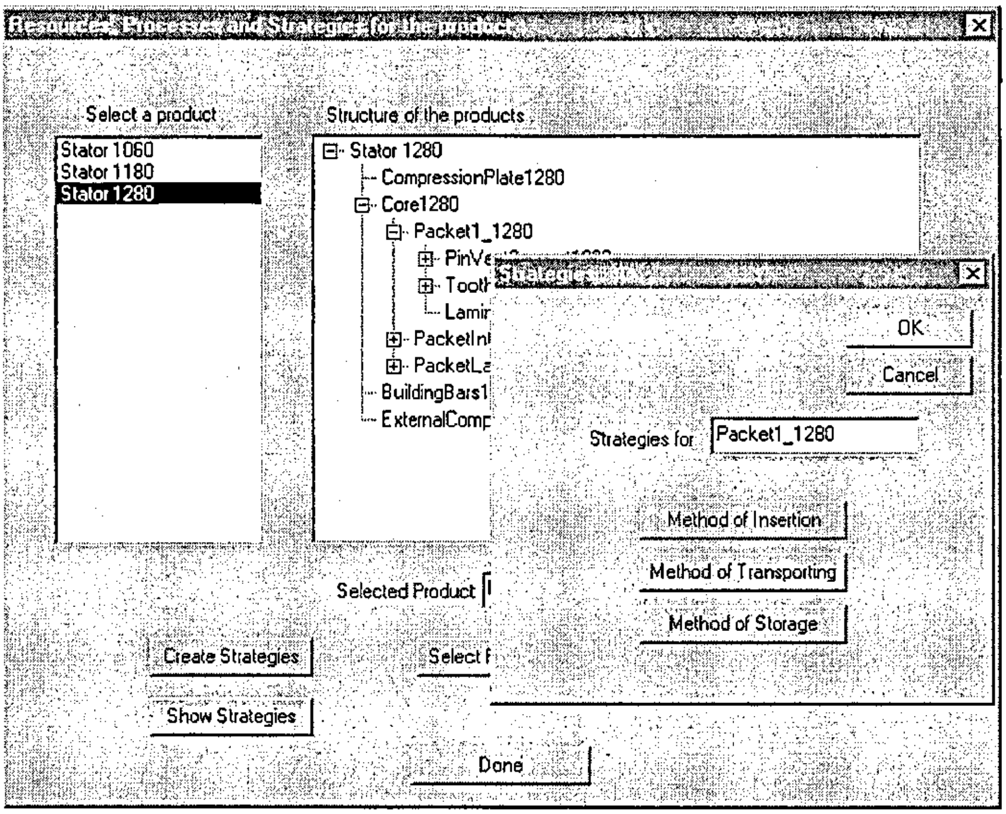



Figure 9.42 Creation of Strategies for a product

Figure 9.43 illustrates the dialog box for the creation of an insertion strategy showing the generic information of the Product to help in the creation of the insertion strategy. If detailed information of the product is required, the user can access it, as shown in Figure 9.12.

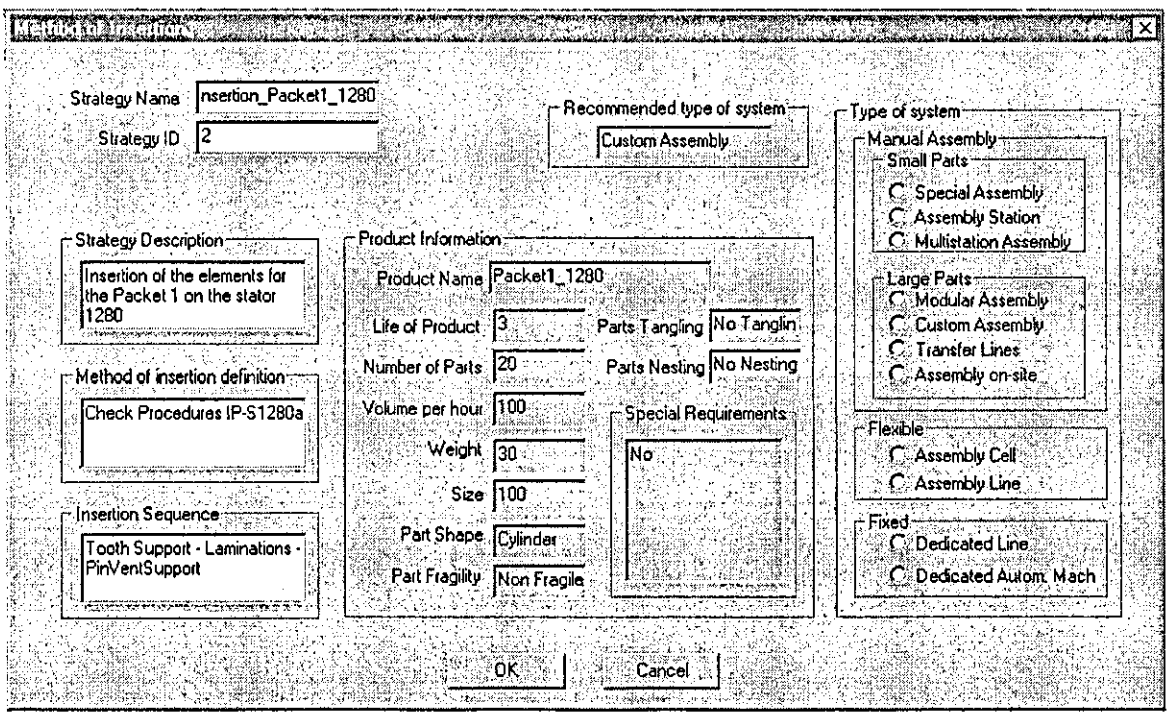



Figure 9.43 Creation of an insertion strategy

Figure 9.44 and Figure 9.45 show the creation of Transporting and Storage strategies, following a format that also reads the information of the product and displays it for guiding the designer.

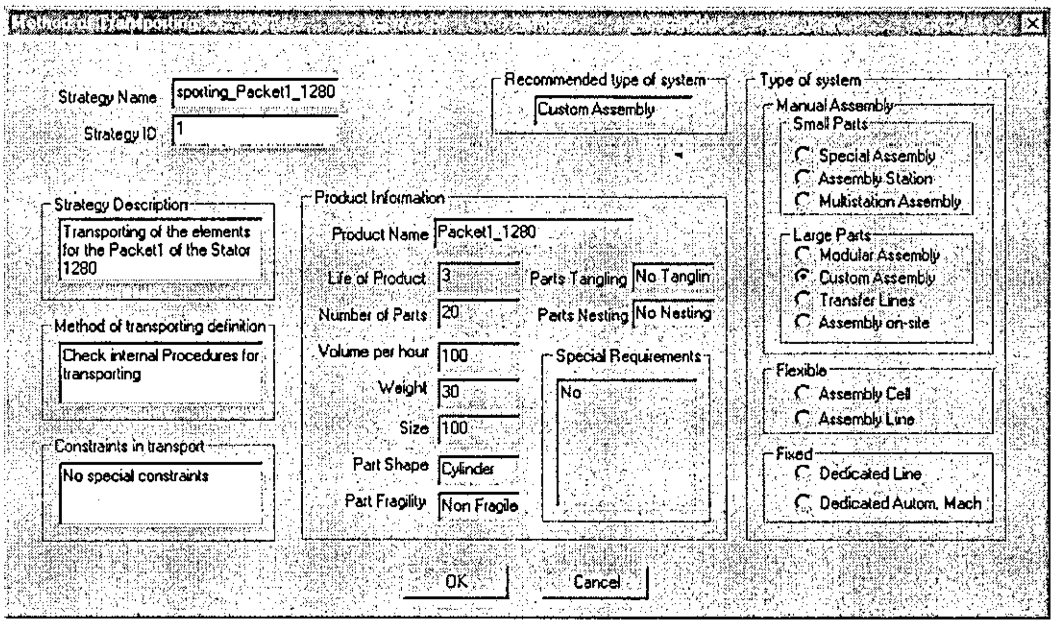



Figure 9.44 Creation of a Transporting Strategy

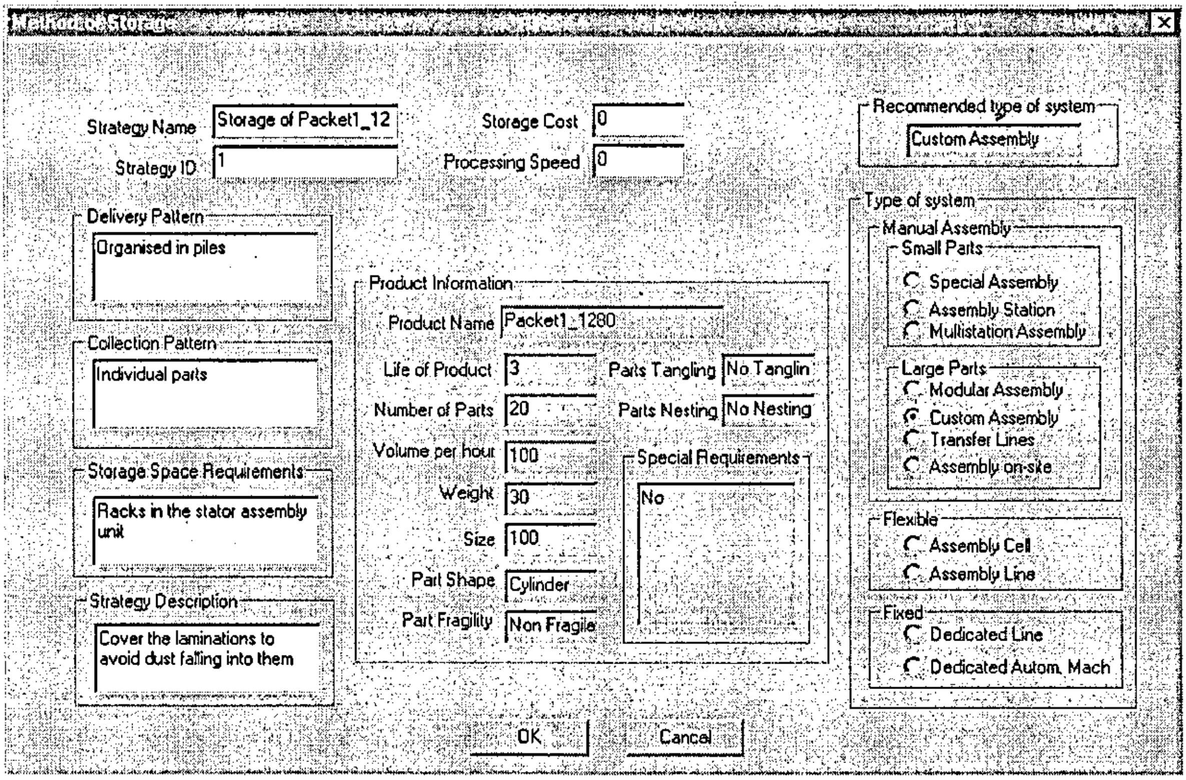



Figure 9.45 Creation of a Storage Strategy

The testing system links strategies to a product when they are created. The information of the strategies linked to a product can be accessed at any time. The strategies are displayed in a single user interface that includes the Transport, Storage and Insertion Strategies. A product can contain several different strategies, so they can be selected from a list and their characteristics are shown as illustrated in Figure 9.46.

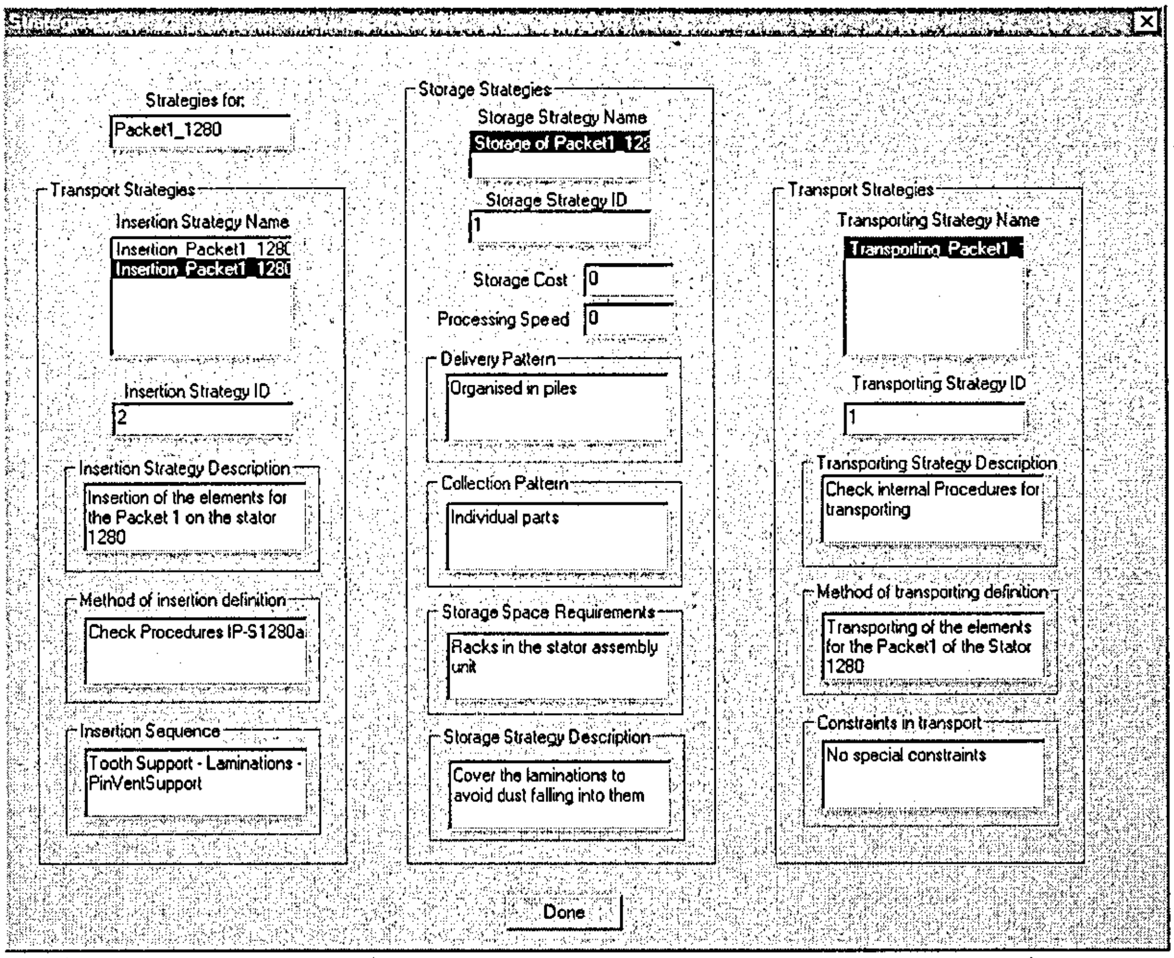



Figure 9.46 Strategies created for the product

### 9.5.6. Case study results

This test has shown that the use of viewpoints in the Product Model allows the multiple viewpoint access to the information stored. The Design View helped to define the connections between parts and the Manufacturing View was used as a link between the Product Model and the Manufacturing Model.

The selection of resources and processes for a product was done through the combined use of the Product Model and the Manufacturing Model.

Strategies were used at different levels of the Facility structure. The selection of the Type of System was done at a Factory level, while the highly product-dependant Assembly Strategies at a Station level were created by using the links existent between the Product Model and the Manufacturing Model.

## 9.6. Structures of the Product Model and Manufacturing Model to Support Design For Assembly and Assembly Process Planning

This section reports a test aimed at the following:

- To illustrate how the information stored in the structures of the Product Model and the Manufacturing Model can be used for supporting assembly related applications
- To show how a simple Design for Assembly application can support the concurrent development process running in the background and providing recommendations to the designer according to the product's characteristics existent in the Product Model.
- *To show how Assembly Process Planning recommendations can be given at the moment of selecting equipment for the assembly of the product.*

When the characteristics of the product are being introduced into the Product Model, some DFA considerations are given to the user. The system assists the user in the definition of symmetries by displaying the dialog boxes shown in Figure 9.47. This definition of symmetries are necessary if the Boothroyd DFMA or the Lucas DFA methods will be applied for evaluating the assemblability of the product in the Detail Design stage of the product development cycle.

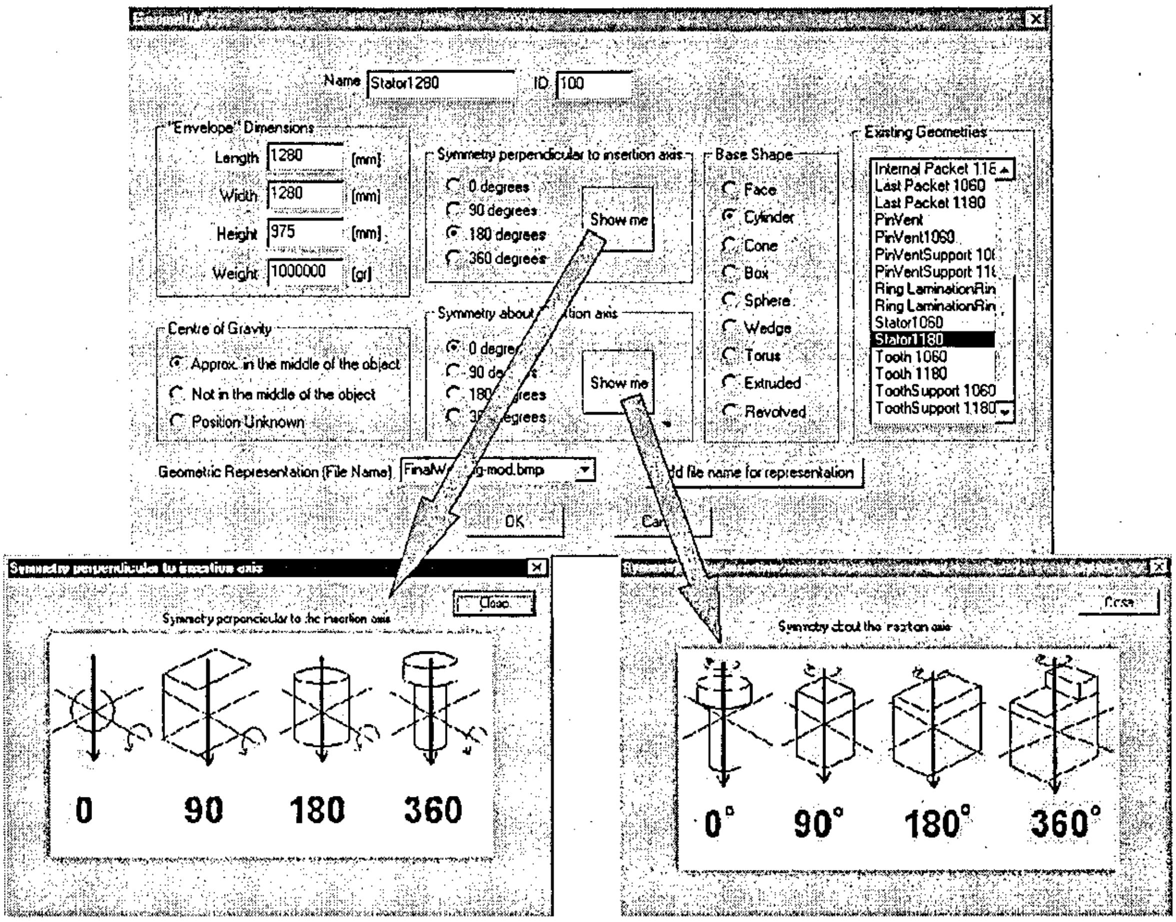



Figure 9.47 Symmetries of the part.

When the geometry has been defined, and before linking that definition with the product, basic DFA recommendations are given, such as the one presented in Figure 9.48.

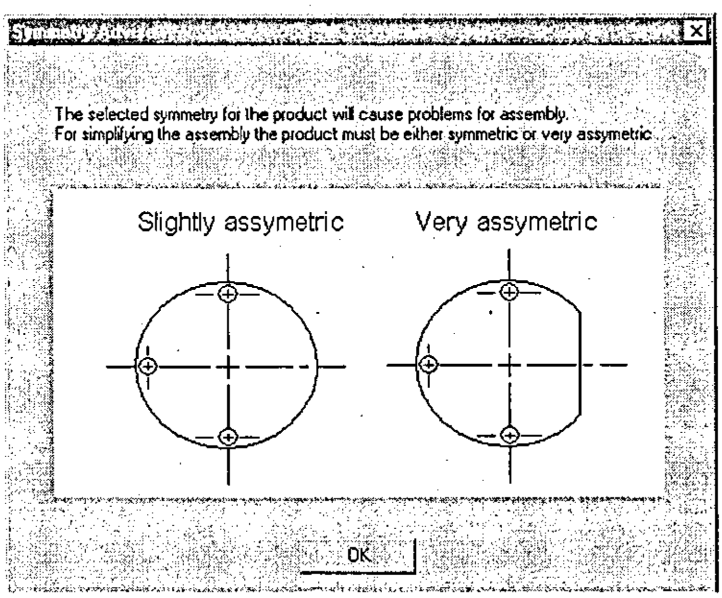



Figure 9.48 DFA recommendation for assymetric parts

Once the dialog box used for linking the characteristics to the product is closed, some DFA questions are asked to determine if the part is necessary or can be eliminated, giving the designer advise about components that are vital for the product and those who are not critical. These questions and the two possible answers are represented in Figure 9.49.

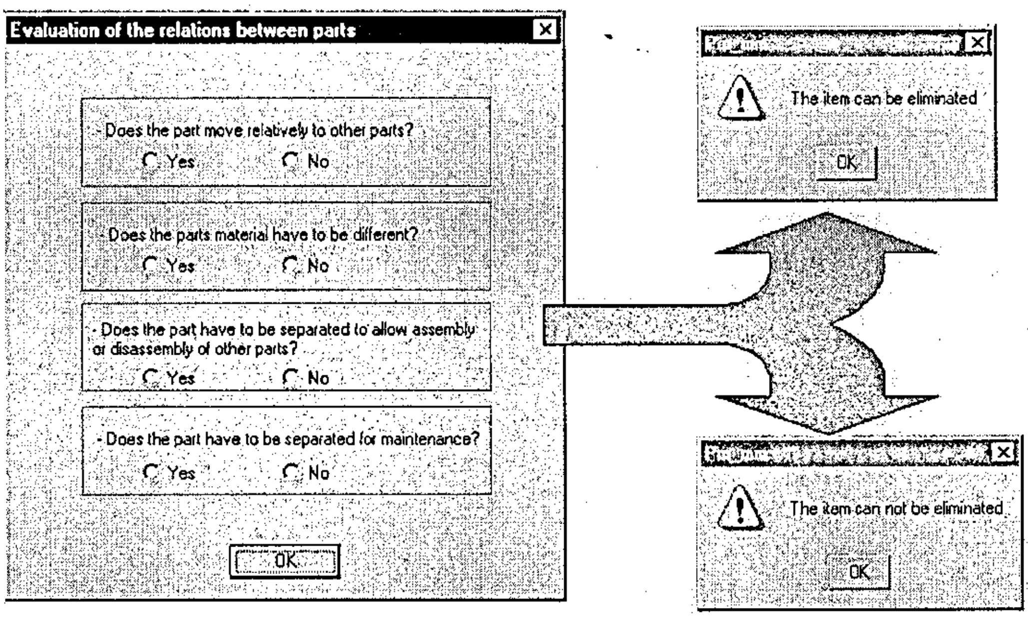



Figure 9.49 Evaluation of the relations between parts

The independent parts are evaluated by the system using DFA parameters that advise the user about possible assembly problems. The implemented criteria detect parts that are big, heavy, very small, fragile or sticky. The results are obtained by analysing the Characteristics of the part. Figure 9.50 shows that the 'Stator 1280' is heavy and big. This analysis can be done for individual parts that are part of an assembly, as shown in the test described in section 9.3.2, and shown in Figure 9.52.

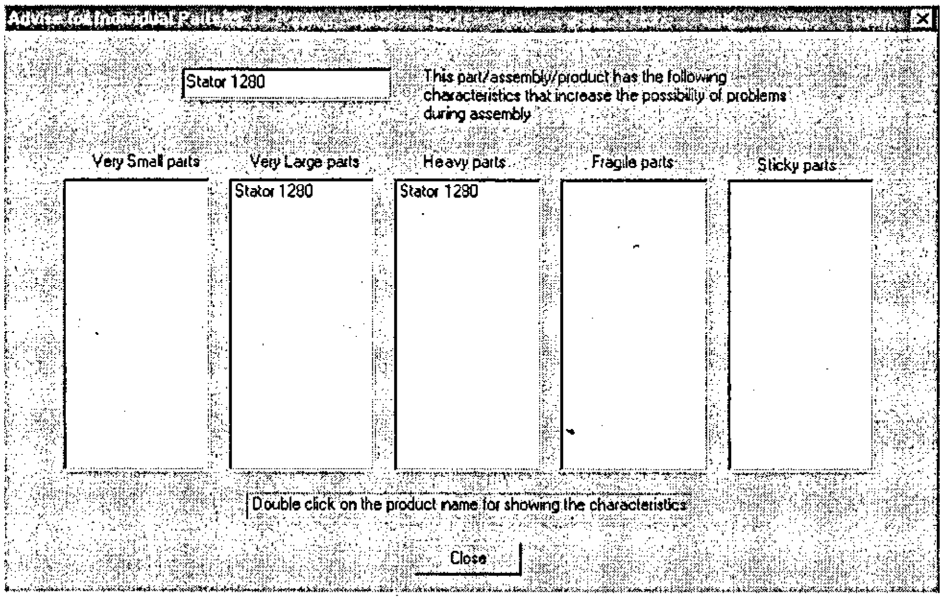



Figure 9.50 Assembly advise for individual parts

The product, subassemblies and components shown in the tree structure can be selected for analysis. This is done at the embodiment design stage of the product life cycle. The dialog box shown in Figure 9.51 shows the options available for this analysis.

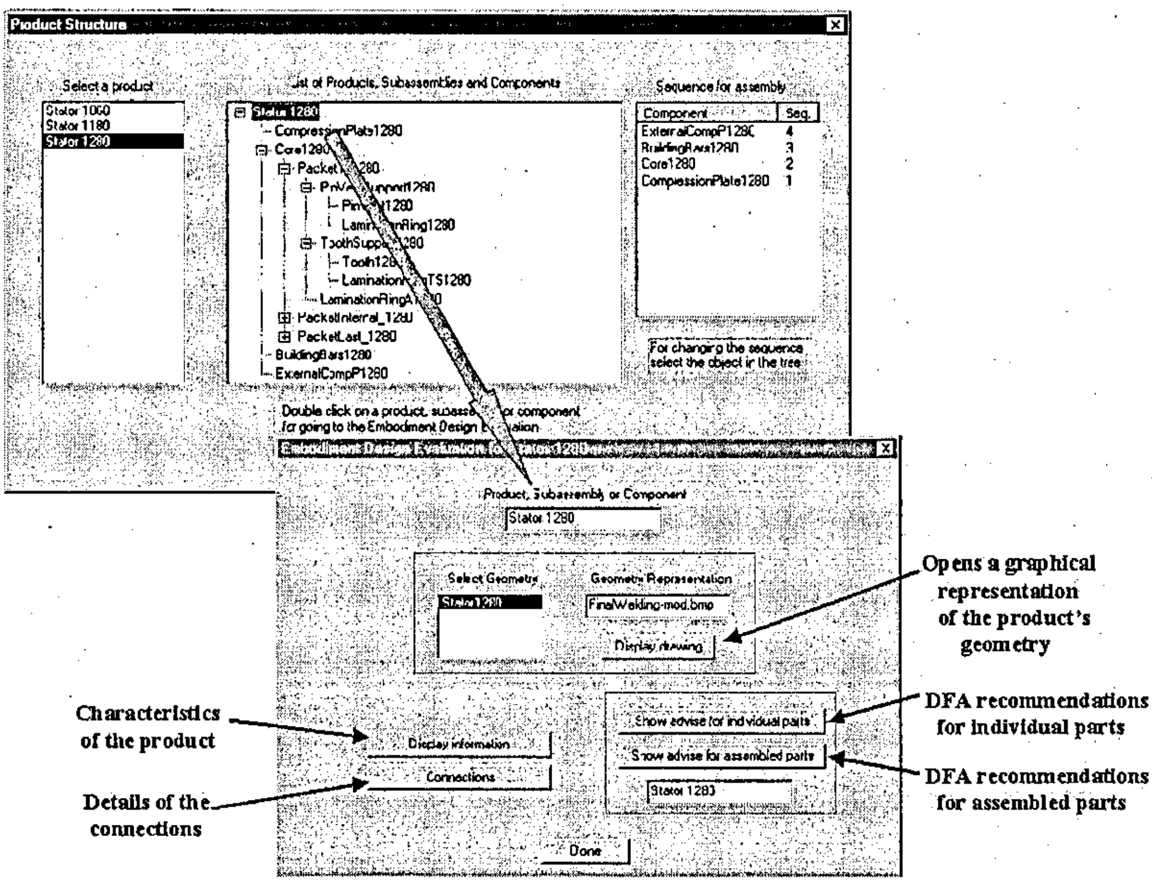



Figure 9.51 Dialog box for the embodiment design stage

The DFA recommendation for individual parts shows the parts that are included in the assembly that have some characteristics that could complicate the assembly. This is shown in Figure 9.52.

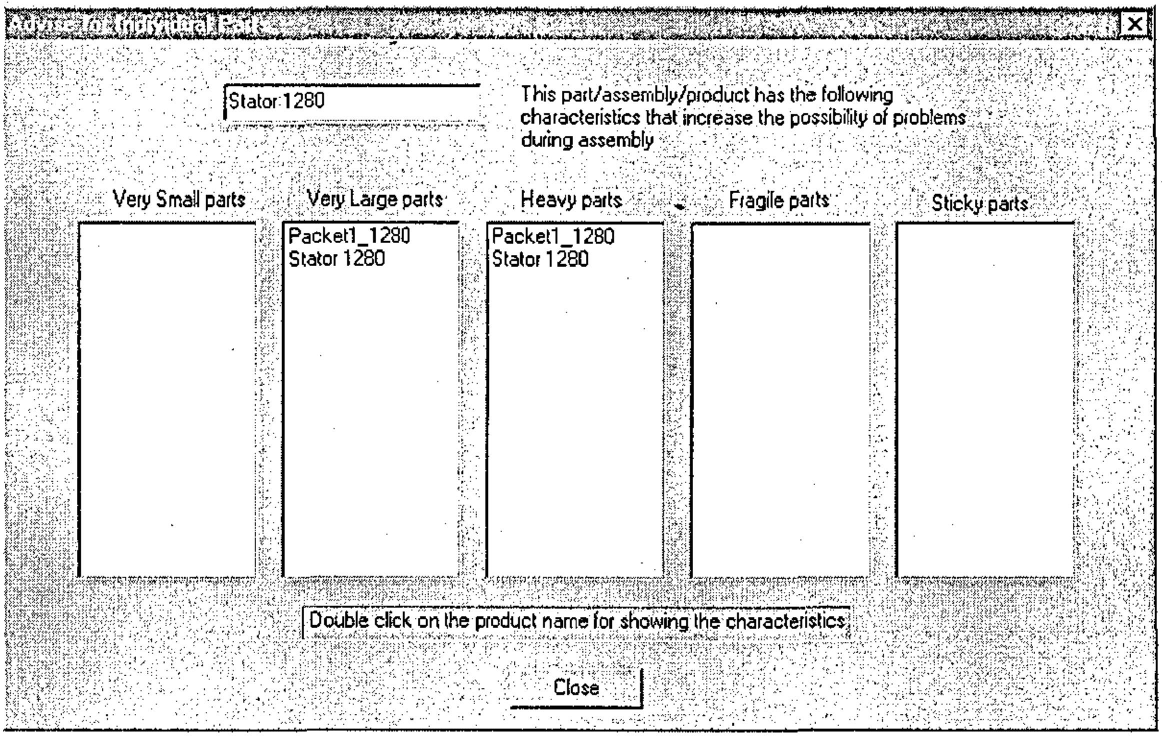



Figure 9.52 Advise for individual parts in an assembly

Figure 9.53 shows the results of the analysis for assembly problems between parts. Here some problems like tight fits, assembly not in a straight line, parts that require holding and temporary connections were selected to show the capability of the Information Models to provide information to the analysis software. Selecting the connection with the problem gives access to details of the problems, and if the part or the connections are redesigned, the information is updated and the problem solved.

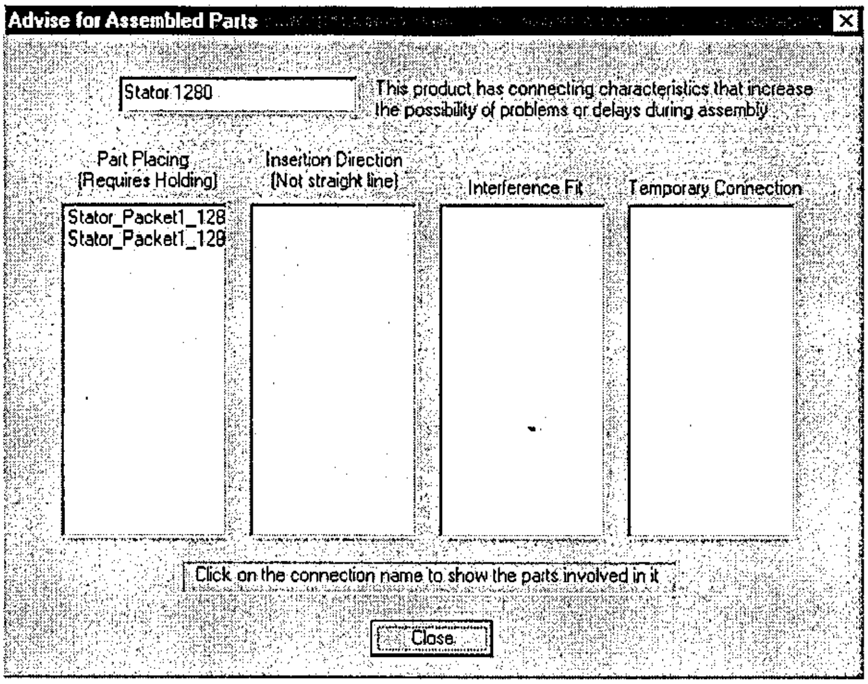



Figure 9.53 Advise for assembled parts

Figure 9.54 shows the process of updating the information for eliminating the problem found when the DFA criteria were applied.

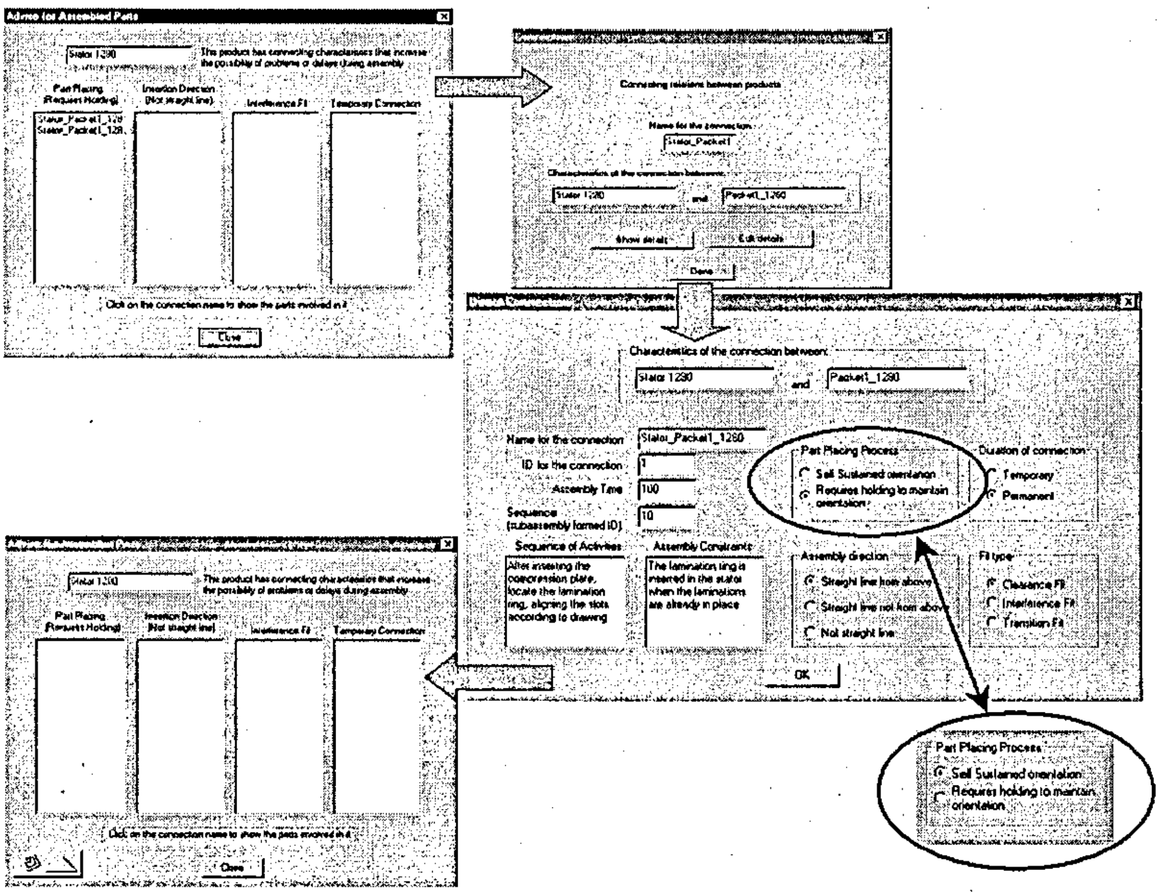



Figure 9.54 Process of editing connections information

The selection of Resources, Processes and Strategies was done in the testing system following the process illustrated in section 9.5. The options given to the user in the lists in the user interface are only those that fulfil the requirements of the parts that are assembled. The Assembly Process Planning recommendations for the selection of Facilities were included into the software, so they run in the background of the system, simplifying their selection by the user.

One of the main functions of Assembly Process Planning is the sequencing of the activities that are necessary for performing the assembly. This test used the information stored in the Product Model and the Manufacturing Model to show the sequence selected by the user for the assembly and the Resources, Processes and Strategies used for these activities. Figure 9.55 shows the dialog boxes with the components, sequence for assembly, and some of the Facilities. The selection of Resources, Processes and Strategies can be completed following the process explained in section 9.5. The information shown in Figure 9.55 can be used for the creation of a Bill of Materials and Assembly Charts.

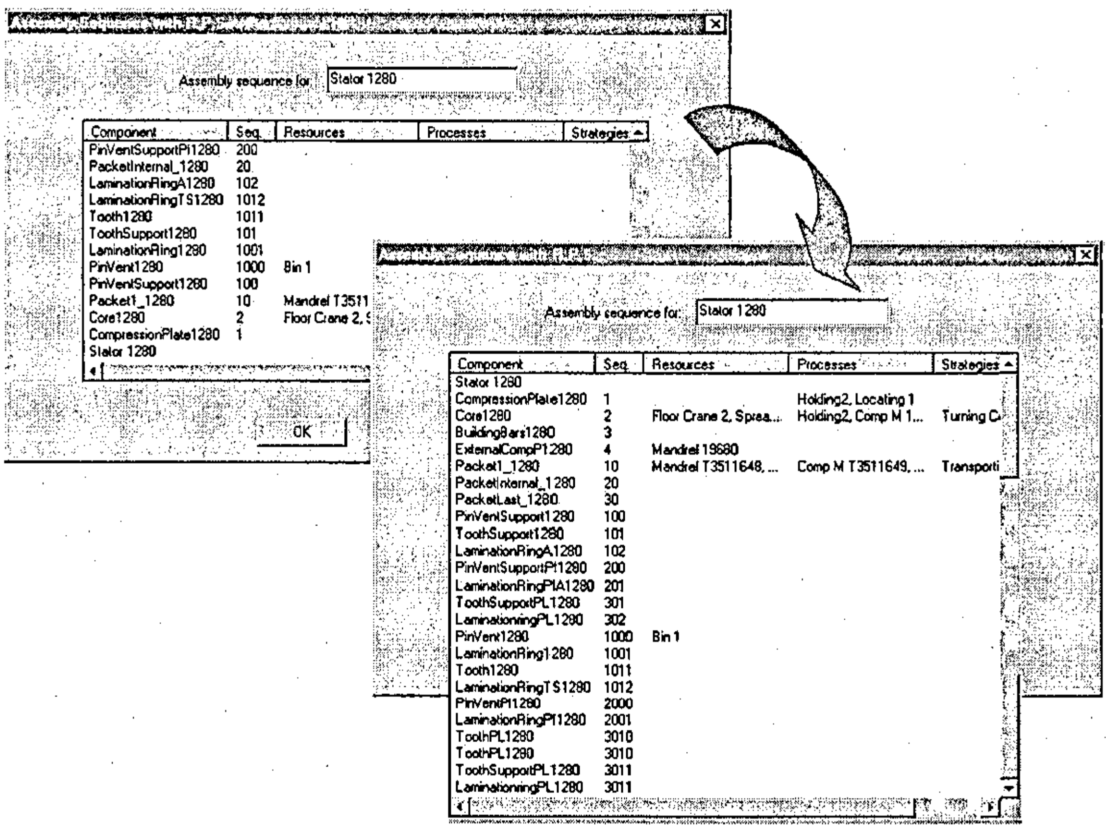


Figure 9.55 Sequence for assembly

### 9.6.1. Case study results

This test has shown how the structures defined for the Product Model and Manufacturing Model are able to support assembly related applications, such as Design For Assembly and Assembly Process Planning.

The recommendations were given by the Design For Assembly application at the conceptual and embodiment stages of the concurrent design process, proving the feasibility of using the proposed structures for the Product and Manufacturing Model for supporting the product development process from the beginning.

The Assembly Process Planning application was able to use the information from the Manufacturing Model to propose which Resources and Processes were the most appropriate for performing the assembly activities.

It was demonstrated that the APP can use the information contained in the Product Model and the Manufacturing Model to check the assembly sequence and the Facilities used in the assembly operations.

## 9.7. Summary

The author has shown, through the testing system, that the structures defined for the Product Model and the Manufacturing Model are capable of providing support, and being a source and repository for the information required in the assembly-related applications Design For Assembly and Assembly Process Planning.

The structure defined for the Product Model allows it to be used as a source and repository of information to support Design For Assembly and Assembly Process Planning through the product development cycle, starting in the conceptual design stage.

The Manufacturing Model structure is able to hold information of the Facilities at different levels, detailing the Resources, Processes and Strategies. This information provides a valuable source of information for the concurrent engineering development team.

The definition of the strategies for assembly is dependent on the information of the product. The interactions between the Product Model and the Manufacturing Model allowed the definition of those strategies and the selection of Resources and Processes for a particular product.

The DFA and APP applications designed to test the structures of the Product Model and Manufacturing Model are able to run in the background of the system, using common input information and delivering different results. The system provided DFA and APP recommendations to the user.

# 10. DISCUSSION, CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

## 10.1. Introduction

The research described demonstrates the way in which Information Models in a Concurrent Engineering environment support assembly-related applications. The structures for the Product Model and the Manufacturing Model were defined and developed for supporting applications such as Design For Assembly and Assembly Process Planning. The integrated approach provided enhanced support for the concurrency of these applications. Tests were performed to show that the structures defined for the Information Models are able to support information interactions between them, and that they can be used as source and repository for the information required by assembly-related applications.

This chapter present a discussion of the approach taken in this research, the conclusions reached, and recommendations for further research.

## 10.2. Discussion

The discussion of the results are presented in separate sections, one for the methodology followed, another for the structures for the Product Model and the Manufacturing Model, and the last for modelling assembly activities.

### 10.2.1. Methodology for Information Capture

The combined integrated methodology applied in this research proved to be useful to model the information from the capture of the activities and requirements of an enterprise through to the creation of the classes and objects required in an object-oriented database.

IDEF0 proved to be an effective method for capturing relevant activities, and it is a good way for communicating with collaborating companies due to the clarity of the relations between activities shown in the diagrams. The construction of IDEF3 process centred descriptions based on the results from the IDEF0 method proved to be simple and reliable.

The elaboration of IDEF3 object centred descriptions helped to check where the Design For Assembly and Assembly Process Planning methods influenced the concurrent design of the product, process and system. These object centred descriptions were very useful as a way of linking top-down and bottom-up approaches for defining the structures required for the information models.

The creation of the Elaboration Diagrams for the Units of Behaviour of the IDEF3 model, although time consuming, provided useful information to define the database schema for the Product Model and the Manufacturing Model. These elaboration diagrams have the limitation that they depend more on the perception of the situation than in formal methodologies to be completed.

UML class diagrams were used to show in detail the classes and their attributes necessary for providing the Product Model and the Manufacturing Model with the capabilities of being a source and a repository of information, the first for the product and the latter for the facilities.

Although the evaluation of methodologies was out of the scope of this thesis, the described structured approach using of IDEF standards and UML methods proved to be valuable for defining the structures of the information models. The author recommends the use of IDEF0/IDEF3 methods before the application of UML methods, because they provide a clear starting point for the definition of the requirements of the system, and the interaction with the environment in which the system will have to operate. A journal paper published about this approach is included as Appendix B.

### 10.2.2. Structures For The Product Model And The Manufacturing Model

The research reported in this thesis provided the definition and design of integrated information structures for Product and Manufacturing Models to support assembly-related activities, particularly for the integration of Design For Assembly and Assembly Process Planning software applications. These structures for the Product Model and the Manufacturing Model extend the capabilities of information models defined by other researchers, which have supported the areas of machining and injection moulding.

Traditionally, assembly-related applications provide support only at the detailed design stage, when the complete information is available. By using the defined structures for the Product and Manufacturing Models, information is made available for analysis at all the development stages. This research contributes to the area of supporting assembly in the early stages of product design by defining Information Models capable of supporting the development team through the stages of concurrent design of the product, process and system. Even though at the early stages the product, process, and system information could be incomplete or inaccurate, it proved to be useful to provide the development team with advice and recommendations for assembly. Furthermore, the structure of the information models proposed allows the aggregation and use of information as soon as the development team defines it.

The structure defined for the Product Model include the Parent, Subassembly, and Component levels, that allow the classification of parts according to their condition in the assembly of the product. These levels allow the complete definition of the characteristics of the components, subassemblies and parent assemblies by using the inheritance property of the object-oriented approach, simplifying the information structure.

The inclusion of Variant, Standard, and Original kind of products in the Product Model gives the possibility to use the information of existing products for the creation of design variants. The possibility of reusing information was proved with the testing software by selecting already defined characteristics and modifying some parameters to be used on a different product. This issue of the reuse of information can be extended in order to be able to support the designer by analysing information from other products.

The work reported in this thesis contributed by extending the structure of the Product Model to be able to support multiple views. This allows the use of the information contained in the Product Model from a new perspective, which provides access to selected information of the product, depending on particular interests. The Design View and Manufacturing View could be extended to support other views.

This research improved the understanding of the interactions between Resources, Processes and Strategies in the Manufacturing Model, allowing the modelling of dependencies between them.

This research extended the interactions between the Product Model and the Manufacturing Model in order to allow information models to support effectively a Concurrent Engineering environment. The testing software uses intensively the possibility of information interaction between the Information Models, allowing the selection of Resources and Processes from the Manufacturing Models using Product characteristics, and the creation of Product dependant Strategies in the Manufacturing Model, which are discussed below.

A significant issue in assembly information modelling, opposed to machining information modelling, it is that assembly strategies are heavily product-dependant. That dependency led to the proposal of an approach that requires an additional relationship between the models that incorporates the creation of product dependant strategies. By creating product dependant strategies in the Product Model and then sending them to the Manufacturing Model, a new way of perceiving the Manufacturing Model is shown.

The author believes that the structure created for capturing stators of electric generators in the Product Model can be used for capturing information about different kind of products, further research can be done in this area. Similarly, the structure defined for the Manufacturing Model can be extended to capture any kind of processes, resources and strategies, maintaining the general structure of the model.

### 10.2.3. Relation of this research with ISO Standards and PDM.

As explained in section 2.3.4, in the ISO there are workgroups developing standards for the standardisation of the information data exchange. The STEP standard is focused on product data representation and exchange, while the MANDATE deals with the representation of manufacturing processes.

The considerations of assembly in the STEP standard were mentioned in section 2.4.5.6, from which it can be seen that the assembly relations remain undefined in the standards. The way of storing the connection information in the structure proposed for the Product Model can be useful for this issue. The geometric representations defined in STEP can be applied for extending the definition given for the Geometry in the Product Model, and by doing so, it would allow the interaction with CAD systems compliant with STEP, simplifying the capture of geometric information in the Product Model.

The underlying structure for the Manufacturing Model defined by the author relates to the MANDATE project, explained in section 2.3.4.2, by proposing a way for capturing Assembly-related Resources, which has not been included yet in the MANDATE project.

While PDM systems provide a way for integrate and manage information that define products across multiple systems and media, this research proposes structures for storing product information in a Product Model, and resources, processes and strategies information in a Manufacturing Model. The storage of Strategies information is nor currently considered by PDM systems.

### 10.2.4. Modelling of Assembly related Activities

The modelling of assembly-related activities and processes during the concurrent development of the product, system, and processes, showed that their influence exists through all the stages of the concurrent development process, and that they require information support, in order to provide effective guidelines and advise to the developers. The models obtained helped to identify the requisites that the structures for the Product Model and the Manufacturing Model had to fulfil.

As a result of the analysis of the influence of DFA and APP through the concurrent development process, it was found that both applications share most of the information required for the analysis even though their results are very different.

DFA and APP applications were created to test the capability for supporting assembly-related application software with the structures defined for the Product Model and the Manufacturing Model. Those applications, although simple, had sufficient capabilities for providing the product development team with information and recommendations about the product and facilities and proved to be able to use effectively the information contained in the Product and Manufacturing Models.

## 10.3. Conclusions

1. An integrated information system based on the use of combined Product and Manufacturing Models was defined to provide an information-sharing environment for a range of assembly activities.

2. It has been shown that the information structures defined for the Product Model and the Manufacturing Model can serve as a source and repository for supporting assembly-related software applications through the concurrent development of the product, system, and process, from conceptual design through to planning.

3. It has been shown that the structure defined for the Product Model is able to support multiple views, in particular the Assembly Design View and the Assembly Manufacturing View, which allow the storing of connection information and the link with the Manufacturing Model information.

4. The inclusion of Variant, Standard and Original kind of products in the Product Model structure, allows the classification of products in families.

5. Within the Product Model, the structure which captures the relation between the Parent, Subassembly and Component levels of the product, is significant in that it allows the capture of their characteristics, the possibility for them to exist in different ranges and the access to its information from different view points.

6. The structure defined for the Manufacturing Model allows it to support assembly processes, resources and strategies, and improved the understanding of the relations and dependencies among them, allowing their selection and definition depending on the characteristics of each other.

7. It has been shown that the structure defined for the Manufacturing Model allows it to support all the stages of the concurrent development of the product, resources, and processes by providing information about the facilities, even if the information is incomplete or under development.

8. It has been shown that having the right structures defined for the Product and Manufacturing Models allow them to have information interaction in order to support the concurrent development of the product, processes, and resources. These interactions allowed the definition of product-dependant assembly strategies in the Manufacturing Model.

9. By providing Design For Assembly and Assembly Process Planning testing software applications with a source of common information, their integration became possible. The use of the Product and Manufacturing Models as the source and repository for assembly-related information proved to be an effective support in a concurrent engineering environment.

10. The use of the methodology followed proved to be effective in the definition of the structures for the Product and Manufacturing Models, and in the design of the case study system. IDEF0 and IDEF3 allowed the understanding and capture of information at high levels, while UML methodology helped in the creation of the detailed attributes for the classes defined.

11. A testing system has been implemented using the object-oriented database ObjectStore© and the Visual C++ programming environment. This system has been explored using real cases from the industrial collaborator to successfully demonstrate the feasibility of supporting assembly-related applications using the structures defined for the Product Model and the Manufacturing Model.

## 10.4. Recommendations for Further Work

This research has defined the structures for the Product Model and the Manufacturing Model. These structures provided the basis for information models, which have the capability of being source and repository of the product and manufacturing facilities information. However, a number of issues were identified in the process of doing this research and are recognised as further work.

1. The testing system, used to capture the stators of electrical motors and the facilities of the collaborating company, was developed using a sufficient subset of the complete structure of the Product Model and the Manufacturing Model defined in chapter 7 of the thesis. The author believes that these Information Models can capture information about other products and facilities, in order to test this capability further implementation should be done applying other sections of the information structures defined.

   - It has been shown that the structure provided for the Product Model is able to capture products classified in Ranges, giving the possibility to constitute product families. However, there is a need for exploring how the use of product families' information, stored in the Product Model can help the development team. Possible trends are the reuse of the design information and the standardisation of components.

- In this research, the use of Design Views and Manufacturing Views proved the convenience of using a multiple view approach in the Product Model. There is a need to investigate how these views can be extended and used to improve the capabilities of accessing information stored in the Product Model.
- The structure defined for the Product Model includes a view based on Feature Based definitions that requires further work to explore its applicability and possibility to interact with feature-based software applications.
- The user performed the population of the characteristics of the product in the testing system. Further work is required to explore the way of interacting with a CAD software system in order to obtain and provide geometric information directly.
- The Resources and Processes in the Manufacturing Model have been captured only at the cell and station levels; further work is necessary to explore how to extend them to higher levels in order to be able to support decisions at those levels.
- Product-dependant strategies were defined and applied, however, there is a need to investigate the way to improve its use in relation to the Resources and Processes in order to provide a better way for influencing their selection.

2. Other area in which further work can be performed is in the development and use of more complete and efficient Design For Assembly and Assembly Process Planning for using the Product Model and the Manufacturing Model. This research defined a simple testing system with the sufficient functionality to test the ideas of the research. The following further research has been identified.

    - The functionality of the Design For Assembly and Assembly Process Planning software applications can be extended in order to be able to provide the user with accurate recommendations, guidelines, and evaluations of the product and the facilities available for assembly, by using the information stored in the Product and Manufacturing Models.

- Further work is necessary to evaluate the possibility of linking commercial DFA software applications, such as the Boothroyd DFMA and the TeamSET with the Information Models.
- The structures for the information models defined in the research could support Assembly Sequence Generation by providing a way of considering not only the constraints and limitations imposed by the geometry of the parts or by the assembly processes, but also taking into account the resources and strategies available. Further research is necessary to explore the use of Assembly Sequence Generation systems with the Information Models.

3. As explained in chapter 5, IDEF0 and IDEF3 models of the concurrent design process were done by the author as part of the methodology followed to find the information requirements imposed by Design For Assembly and Assembly Process Planning to the Information Models. These models were tested using the information provided by Alstom Electrical Machines about their rotors and stators. However, it is necessary to validate their generality by applying them to a wider variety of products.

# 11. REFERENCES

- Ames, A. L.,Calton, T. L.,Jones, R. E., *et al.*, 1995, Lessons Learned from a Second Generation Assembly Planning System. *IEEE International Symposium on Assembly and Task Planning*, Pittsburgh, PA, USA, Aug 10-11 1995 , pp. pp. 41-47.
- Andreasen, M. M., 1988, Design for Assembly expanded to CIM. *9th International Conference on Assembly Automation*, London, UK, March, 1988 , pp. 119-128.
- Andreasen, M. M., 1994, Modelling - The Language Of The Designer. *Journal of Engineering Design* **5**(2), 103-115.
- Andreasen, M. M. and Ahm, T., 1986, The relation between product design, production, layout and flexibility. *7th International Conference on Assembly Automation*, Zurich, Switzerland, , pp. 161-172.
- Ashworth, M.,Bloor, M. S.,Mc Kay, A., *et al.*, 1996, Adopting Step For In-Service Configuration Control. *Computers in Industry* **31**, 235-253.
- Baker, R. P. and Maropoulos, P. G., 2000, An architecture for the vertical integration of tooling considerations from design to process planning. *Robotics and Computer-Integrated Manufacturing* **16**(2), 121-131.
- Baldwin, D. F.,Abell, T. E.,Lui, M. C. M., *et al.*, 1991, An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products. *IEEE Transactions on Robotics and Automation* **7**(**1**), 78-94.
- Baxter, J. E. 1994, *A Functional Data Model for Assemblies,* PhD Thesis, (Leeds, England: Leeds University).
- Belson, D., 1994. Concurrent Engineering. *Book* Handbook of design, manufacturing and automation, (New York: Wiley), pp. 25-34.
- Bloor, M. S. and Owen, J., 1991, CAD/CAM product data exchange: the next step. *Computer Aided Design* **23**(4), 237-242.
- Boothroyd, G. and Alting, L., 1992, Design For Assembly And Disassembly. *Annals of the CIRP* **41**(2).
- Boothroyd, G.,Dewhurst, P., and Knight, W. 1994, *Product Design for Manufacture and Assembly* (New York: Marcel Dekker, Inc.).
- Borja, V. 1997, *Redesign Supported by Data Models with Particular Reference to Reverse Engineering,* PhD Thesis, (Loughborough: Loughborough University).

- Boubekri, N. and Nagaraj, S., 1993, An Integrated Approach for the Selection and Design of Assembly Systems. *Integrated Manufacturing Systems* **4**(1), 11-17.
- Bradley, H. D. and Maropoulos, P. G., 1997, A Relation Based Product Model For Computer Supported Early Design Assessment. *13th Conference on Computer Aided Production Engineering (CAPE-97)*, Poland, June, , pp. 57-64.
- Bralla, J. G., 1986. Mechanical Assemblies. *Book* Handbook of Product Design for Manufacturing, (New York: McGraw-Hill), pp. 7.79-7.100.
- BSI, B. S. I., 1975. BS 5191: Glossary of Production planning and control terms.
- Calton, T. L., 1999, Advancing Design-for-Assembly, The Next Generation in Assembly Planning. *IEEE International Symposium on Assembly and Task Planning*, Porto, Portugal, July, 1999 , pp. pp. 57-68.
- Chen, W. J. and Wallace, K. M., 1993, Manufacturing Information for Designers. *International Conference on Engineering Design, ICED '93.*, The Hague, , pp. 1111-1114.
- Cheung, W. M., J. Zhao, Dorador, J.M. , Young, R. I. M., 2000, Linking product design and manufacturing capability through a manufacturing strategy representation. *16th International Conference on Computer-Aided Production Engineering (CAPE2000)*, Edinburgh, UK, 7-9th August, 2000 , pp. 615-622
- CIRP, 1990, Nomenclature and Definitions for Manufacturing Systems (English Language Version). *Annals of the CIRP* **39**(2), 735-742.
- Colquhoun, G. J. and Baines, R. W., 1991, A Generic IDEF0 Model Of Process Planning. *International Journal Production Research* **29**(11), 2239-2257.
- Cunha, P. F.,Mesquita, R. M., and O'Kane, J. F., 1999, A Methodology for Evaluation of Process and Production Planning. *15th International Conference on Computer-Aided Production Engineering CAPE'99*, , pp. 157-164.
- Connacher, H. I.,Jayaram, S., and Lyons, K., 1995, Virtual Assembly Design Environment. *Computers in Engineering Conference and the Engineering Database Symposium, ASME 1995*, , pp. 875-885.

- Costa, C. A. 2000, *Product Range Models in Injection Mould Tool Design*, PhD, (Loughborough, Leicestershire: Loughborough University).
- Cunha, P. F.,Mesquita, R. M., and O'Kane, J. F., 1999, A Methodology for Evaluation of Process and Production Planning. *15th International Conference on Computer-Aided Production Engineering CAPE'99*, , pp. 157-164.
- De Fazio, L. and The Design Automation Committee, 1993, A Prototype of Feature-Based Design for Assembly. *Journal of Mechanical Design, ASME* **115**, 723-734.
- De Fazio, T. and Whitney, D. E., 1987, Simplified Generation of All Mechanical Assembly Sequences. *IEEE Journal of Robotics and Automation, RA-3* **6**, 640-658.
- De Fazio, T. L.,Rhee, S. J., and Whitney, D. E., 1997, A Design-Specific Approach to Design for Assembly (DFA) for Complex Mechanical Assemblies. *Proceedings of the 1997 IEEE International Symposium on Assembly and Task Planning*, Marina del Rey, CA., August, 1997 , pp. 152-157.
- Delchambre, A. 1996, *CAD Method for Industrial Assembly* (Chichester: John Wiley & Sons).
- Delchambre, A.,Fan, I. S.,Wafflard, A., *et al.*, 1996, Concurrent Engineering Environment for Product Design, Assembly and Resource Planning. *Sixth International FAIM Conference: Flexible Automation and Intelligent Manufacturing 1996*, Atlanta, USA, , pp. 322-331.
- Delchambre, A.,Gaspart, P., and Wafflard, A., 1989, An Automatic Assembly Planning Approach. *Proceedings of the 10th International Conference on Assembly Automation*, Kanazawa, Japan, October, 1989 , pp. 319-326.
- Deneux, D., 1997, Introduction to Assembly Features, An Illustrated Synthesis Methodology. *Journal of Intelligent Manufacturing* **10**(1), 29-39.
- Dorador, J. M. and Young, R. I. M., 1999, Information Models to Support the Interactions between Design For Assembly and Assemly Process Planning. *International Symposium on Assembly and Task Planning*, Porto, Portugal, , pp. 39-44.
- Dorador, J. M. and Young, R. I. M., 2000, Application of IDEF0, IDEF3 and UML methodologies in the creation of information models. *International Journal of Computer Integrated Manufacturing* **13**(5), 430-445.

- Doumeingts, G., 1998. GIM, Grai Integrated Methodology. *Book* Handbook of Life Cycle Engineering, (Dordrecht, The Netherlands: Kluwer Academic Publishers), pp. 227-288.
- Ellis, T. I. A., 1993, Glossary of Terms. Loughborough, Loughborough University, University of Leeds. Loughborough.
- ElMaraghy, K. F.,Zhang, K. F., and Chu, H., 1993, A Function-Oriented Modeler Prototype. *DE-Vol 53, Design for Manufacturanility ASME.*
- Eng, T. H.,Ling, Z. K.,Olson, W., *et al.*,, 1999, Feature-based assembly modeling and sequence generation. *Computers and Industrial Engineering* **36**, 17-33.
- Fowler, J. 1995, *STEP for Data Management Exchange and Sharing* (Great Britain: Technology Appraisals).
- Fox, J., 1995, Managing design for a quality product,. *Assembly Automation* **15**(1), 12-14.
- Gottschlich, S.,Ramos, C., and Lyons, D., 1994, Assembly and Task Planning: A Taxonomy. *IEEE Robotics and Automation Magazine* **1**(3), 4-12.
- Grewal, S., 1997, Integrated Part and Assembly Planning. *Proceedings of the 1997 IEEE International Symposium on Assembly and Task Planning, Marina del Rey, CA, August, 1997*, pp. 36-41.
- Gupta, R. and Zeltzer, D., 1995, Prototyping and Design for Assembly Analysis using Multimodal Virtual Environments. *Computers in Engineering Conference and the Engineering Database Symposium ASME 1995.*, , pp. 887-903.
- Gupta, S. K., W.C., R.,Das, D., *et al.*, 1997, Automated Manufacturability Analysis: A Survey,. *Research in Engineering Design* **9**, 168-190.
- Harding, J. A. 1996, *A knowledge representation model to support concurrent engineering team working,* PhD, (Loughborough: Loughborough University).
- Hartley, J. R. 1992, *Concurrent Engineering Productivity* ( U.S.A.: Press).
- Henson, B., 1993, Assembly Representation Within a Product Data Framework. Leeds, UK, University of Leeds & Loughborough University. Leeds, UK.
- Henson, B., 1995a, Assembly Modelling Workpackage Final Report. Leeds, UK, University of Leeds & Loughborough University. Leeds, UK.
- Henson, B., 1995b, Review Of Assembly Modelling. Leeds, UK, University of Leeds & Loughborough University. Leeds, UK.

- Herbertsson, J. and Johansson, C., 1993, "Are Design For Assembly Methods Necessary in Integrated Product Design Projects?". *International Conference on Engineering Design, ICED*, The Hague., , pp. 1097-1102.
- Hird, G., Swift,K.G., B.,R., S., *et al.*, 1988, Possibilities for integrated design and assembly planning. *9th International Conference on Assembly Automation*, London, UK, March , pp. 155-166.
- Homem de Mello, L. S. and Sanderson, A. C., 1990, And/Or Graph Representation of Assembly Plans. *IEEE Transactions on Robotics and Automation* **6**(2), 188-199.
- Hsu, W., Lee, C.S.G., Su, S.F., 1993, Feedback approach to design for assembly by evaluation of assembly plan. *Computer Aided Design* **25**(7), 395-410.
- Hsu, W., Lim, A., Lee, C.S.G., 1996, Conceptual Level Design for Assembly Analysis Using State Transitional Approach. *Proceedings of the 1996 13th International Conference on Robotics and Automation*, , pp. 3355-3360.
- Huang, C. C. and Kusiak, A., 1998, Manufacturing Control with a Push-Pull Approach. *International Journal of Production Research* **36**(1), 251-275.
- International Standards Organisation, 1997, MANDATE, ISO TC184/SC4/WG8 N147, http://www.nist.gov/sc4/mandate/parts/part001/current/part1.doc
- Jakiela, M. J. and Palambros, P., 1989, Design and implementation of a prototype CAD system. *ASME Journal of Mechanisms, Transmissions and Automation in Design* **111**(2).
- Jorysz, H. R. and Vernadat, F. B., 1990, CIM-OSA Part 1: Total enterprise modelling and function view. *International Journal of Computer Integrated Manufacturing* **3**(3), 144-180.
- Kaufman, S. G.,Wilson, R. H.,Jones, R. E., *et al.*, 1996, The Archimedes 2 Mechanical Assembly Planning System. *IEEE Intl. Conf. on Robotics and Automation*, , pp. pp. 3361-8.
- Kim, G. J. and Bekey, G. A., 1994, Design-For-Assembly (DFA) by Reverse Engineering. *Artificial Intelligence in Design*, 717-734.
- Kim, G. J.,Bekey, G. A., and Goldberg, K. Y., 1992, A Shape Metric for Design for Assembly. *IEEE International Conference on Robotics and Automation*, , pp. 968-973.

- Kim, G. J.,Lee, S., and Bekey, G. A., 1996, Interleaving Assembly Planning and Design. *IEEE Transactions on Robotics and Automation* **12**(2), 246-251.
- Kim, G. J. and Szykman, S., 1996, Combining Interactive Exploration and Optimization for Assembly Design. *1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference*, Irvine, California, USA, August 18-22, 1996 , pp.
- Kim, G. J. and Szykman, S., 1998, Combining interactive exploration and optimization for assembly design. *Transactions of the ASME, Journal of Mechanical Design* **120**, 24-31.
- Kjellberg, T. and Schmekel, H., 1992, product modelling And 'Information-Integrated' Engineering Systems. *Annals of the CIRP* **41**(1).
- Krause, F. L.,Kimura, F.,Kjellberg, T., *et al.*, 1993, product modelling. *Annals of the CIRP* **42**(2), 695-706.
- Kusiak, A. and Zakarian, A., 1996, Reliability and risk analysis of process models. *American Society of Mechanical Engineers, Materials Division (Publication)*(MDv 69-2, (1995)), p 959-980.
- Leaney, P. G. and Wittenberg, G., 1992, Design for Assembling. *Assembly Automation* **12**(2), 8-17.
- Lee, K. and Gossard, D. C., 1985, A hierarchical data structure for representing assemblies: Part 1. *Computer Aided Design* **17**(1), 15-19.
- Li, R. K. and Huang, C. L., 1992, Assembly Code Generation from a Feature-Based Geometric Model. *International Journal of Production Research* **30**(3), 627-646.
- Lieberman, L. I. and Wesley, M. A., 1977, AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly. *IBM Journal of Research and Development* **21**(4), 321-333.
- Lim, S. S.,Lee, I. V. H.,Lim, L. E. N., *et al.*, 1995, Computer-Aided concurrent design of product and assembly processes: a literature review. *Journal of Design and Manufacturing* **5**, 67-88.
- Liu, T. H. and Fischer, G. W., 1993, Developing Feature-Based Manufacturing Applications Using Pdes/Step. *Concurrent Engineering: Research and Applications* **1**, 39-50.
- Liu, D. T. and Xu, X. W., 2001, A review of web-based product data management systems. *Computers in Industry* **44**(3), 251-262.

- Lotter, B., 1986, Planning and implementation of flexible assembly cells. *7th International Conference on Assembly Automation*, Zurich, Switzerland, , pp. 1-9.
- MANDATE, 1998, THE MANDATE PROJECT, *http://www.nist.gov/sc4/www/mandate.htm*
- Mantyla, M., 1990, A Modelling System For Top-Down Design Of Assembled Products. *IBM Journal of Research Development* **34**(5), 636-659.
- Maropoulos, P. G., 1995, A novel process planning architecture for product-based manufacture. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* **204**(B4), 267-276
- Martin-Vega, L. A. and Brown, H. K., 1995, Industrial Perspective On Research Needs And Opportunities In Manufacturing Assembly. *Journal of Manufacturing Systems* **14**(1).
- Mascle, C., 1999, Feature-based assembly model and multi-agents system structure for computer-aided assembly. *International Symposium on Assembly and Task Planning, 1999. (ISATP '99)*, Porto, Portugal, , pp. 8 -13.
- Mayer, R. J.,Menzel, C. P.,Painter, M. K., *et al.*, 1995, Information Integration for Concurrent Engineering (IICE) IDEF3 Process Description Capture Method Report,
- McKay, A., Bloor, M.S., 1991, The Role Of Product Models In Effective CADCAM. *Proceedings of the Institution of Mechanical Engineers- Effective CADCAM '91*, 6-7 November 1991 , pp. 113-119.
- McKay, A.,Bloor, M. S., and de Pennington, A., 1996, A Framework For Product Data. *IEEE Transactions on knowledge and data engineering* **8**(5), 825-838.
- Miyakawa, S. and Ohashi, T., 1986, The Hitachi Assemblability Evaluation Method. Japan, Production Engineering Research Laboratory Hitachi Ltd. Japan.
- Miyakawa, S.,Ohashi, T., and Iwatea, M., 1990, The Hitachi New Assemblability Evaluation Method (AEM). *Transactions of the North American Manufacturing Research Institution of SME*, 352-359.
- Mo, J.,Cai, J.,Zhang, Z., *et al.*, 1999, DFA-oriented assembly relation modelling. *International Journal Computer Integrated Manufacturing* **12**(3), 238-250.

- Molina, A. 1995, *A Manufacturing Model to Support Data-Driven Applications for Design and Manufacture,* PhD Thesis, (Loughborough, England: Loughborough University of Technology).
- Molina, A.,Al-Ashaab, A. H.,Ellis, T. I. A., *et al.*, 1994, A Review Of Computer Aided Simultaneous Engineering Systems. *Research in Engineering Design Journal* **7**(1), 38-65.
- Molina, A.,Ellis, T. I. A.,Young, R. I. M., *et al.*, 1995, Modelling Manufacturing Capability to Support Concurrent Engineering. *Concurrent Engineering: Research and Applications* **3**(1), 29-42.
- Molloy, E.,Yang, H., and Browne, J., 1991, Design for Assembly within Concurrent Engineering. *Annals of the CIRP* **40**(1), 107-111.
- Molloy, E.,Yang, H., and J., B., 1993, Feature-based modelling in design for assembly. *International Journal of Computer Integrated Manufacturing* **6**(1-2), 119-125.
- MOSES, 1994a, MOSES Seminar: Information Systems for Competitive Manufacture. , Loughborough University , UK.
- MOSES, 1994b, MOSES: THE ASSEMBLY DATA WORK PACKAGE, http://leva.leeds.ac.uk/www_moses/assem_wp.html
- MOSES, R. g., 1992, MOSES: Manufacturing Model & Product Model, http://leva.leeds.ac.uk/www_moses/mm.html
- Nevins, J. L. and Whitney, D. E., 1978, Computer-controlled Assembly. *Scientific American* **238**(2), 62-74.
- NIST, 1993, Integration Definition for Function Modeling (IDEF0), http://www.idef.com/Downloads/free_downloads.html
- NIST, 1997, The STEP Project, http://www.nist.gov/sc4/
- Nnaji, B. O., 1997, Pro-Mod-S: A Product Design & Fabrication System for Sheet Metal Products, http://www.ecs.umass.edu/mie/labs/arl/
- Nnaji, B. O.,Liu, H. C., and Rembold, U., 1993, A Product Modeller For Discrete Components. *Int J. Prod. Res.* **Vol 31**(9), 2017-2044.
- Nof, S. Y., Wilhelm, W.E., Warnecke, H.J., 1997, *Industrial Assembly* (London: Chapman & Hall).
- ObjectDesign 1998, *ObjectStore Management Release 5.1* (Burlington: ObjectDesign, Inc.).
- O'Grady, P. and Oh, J. S., 1991, A Review of Approaches to Design for Assembly. *Concurrent Engineering* **3**, 5-12.

- Oh, J. S.,O'Grady, J. S., and Young, R. E., 1995, A Constraint Network Approach to Design for Assembly. *IIE Transactions* **27**, 72-80.
- Owen, T., 1985. The Assembly Process, Product and Design for Assembly. *Book* Assembly with Robots, (Englewood Cliffs, New Jersey: Prentice-Hall Inc.), pp.
- Pels, H. J., 1996, Product and Process Data Modelling: Introduction. *Computers in Industry* **31**, 191-194.
- Pham, D. T. and Dimov, S. S., 1999, A system for automatic extraction of feature-based assembly information. *Institution of Mechanical Engineers* **213(b)**, 97-101.
- Philpotts, M., 1996, An introduction to the concepts, benefits and terminology of product data management. *Industrial Management & Data Systems* **96**(4), 11-17
- Poli, C.,Graves, R., and Groppetti, R., 1986. Rating Products for Ease of Assembly. Machine Design.
- Prasad, B., 1996, Toward A Functional Design Of A Concurrent Information Modeling System. *CMSE, Computer Modeling and Simulation in Engineering* **1**, 7-29.
- Pye, A., 1997, Seeing is Believing. *Manufacturing Computer Solutions*, 23-26.
- Quatrany, T. 1998, *Visual Modeling with Rational Rose and UML* (Reading, Massachussets: Addison-Wesley).
- Rampersad, H. K., 1995a, A Case Study in the Design of Flexible Assembly Systems. *International Journal of Flexible Manufacturing Systems* **7**, 255-286.
- Rampersad, H. K., 1995b, Concurrent design of product process and robotic assembly system. *Assembly Automation* **15**(1), 21-28.
- Rampersad, H. K., 1995c, An Integral Assembly Model. *Journal of Intelligent Manufacturing* **6**, 41-51.
- Rampersad, H. K., 1996, Integrated and Assembly Oriented Product Design. *Integrated Manufacturing Systems* **7**(6), 5-15.
- Regli, W. C., Gaines, D.M., 1997, A Repository for design, process planning and assembly. *Computer Aided Design* **29**(12), 895-905.
- Requicha, A. A. G. and Voelcker, H. B., 1983, Solid Modelling: Current Status and Research Directions. *IEEE Computer Graphics and Applications*(Oct. 1983), 25-37.

- Richter, M.,Rieth, D., and Slinger, M., 1989, Integrated Design and Assembly Planning (IDAP). *10th International Conference on Assembly Automation*, Kanazawa, Japan, October , pp. 327-334.
- Rosario, L. M. and Knight, W. A., 1989, Design for Assembly Analysis: Extraction of Geometric Features from a CAD System Data Base. *Annals of the CIRP* **38**(1), 13-16.
- Sackett, P. J. and Holbrook, A. E. K., 1988, DFA as a primary process decreases design deficiencies. *Assembly Automation* **8**(3), 137-140.
- Schlenoff, C.,Knutilla, A., and Ray, S., 1996, Unified Process Specification Language: Requirements for Modeling Process National Institute of Standards and Technology, http://www.mel.nist.gov/msidlibrary/summary/9660.html
- Schlenoff, C.,Knutilla, A., and Ray, S., 1997, Requirements for Modeling Manufacturing Process: A New Perspective. *Design Engineering Conferences*, Sep , pp.
- Seidel, U. A. and Swift, K. G., 1989, Operation Networks for Coordinated DFA and Sequence Planning. *10th International Conference on Assembly Automation*, Kanazawa, Japan, October, 1989 , pp. pp. 539-546.
- Shah, J. J. and Rogers, M. T., 1993, Assembly Modeling as an Extension of Feature-Based Design. *Research in Engineering Design* **5**(3), 218-237.
- Shah, J. J. and Tadepalli, R., 1992, Feature Based Assembly Modelling. *Computers in Engineering, ASME* **1**, 253-260.
- Shimizu, Y. and Nishiyachi, K., 1996, The Development of Computer Aided Process Planning (CAPP) for Assembly. *Japan/USA Symposium on Flexible Automation*, , pp. 811-814.
- Simpson, T. W., Bauer, M.D., Allen, J.K., Mistree F., 1995, Implementation of DFA in Conceptual and Embodiment Design using Decision Support Systems. *Design Engineering Technical Conferences, ASME*, 1995 , pp. 119-126.
- Sodhi, R. and Turner, J. U., 1993, Towards modelling of assemblies for product design. *Computer Aided Design* **26**(2), 85-97.
- Sturges, R. H. and Kilani, M. I., 1992a, Towards an integrated design for an assembly evaluation and reasoning system. *Computer Aided Design* **24**(2), 67-79.

- Suarez, R. and Lee, S., 1997, The Concept of "Difficulty" in Mating Operations. *The 1997 IEEE International Symposium on Assembly and Task Planning*, Marina del Rey, CA.,, August, 1997. , pp. 176-181.
- Syan, C. S. and Menon, U. 1994, *Concurrent Engineering, concepts, implementation and practice* ( London: Chapman & Hall).
- Taleb-Bendiab, A.,Oh, V.,Sommerville, I., *et al.*, 1993, Semantic Modelling Of Enigneering Designs. *Journal of Engineering Design* **4**(1).
- Taylor, G. D., 1997, Design for global manufacturing and assembly. *IIE Transactions* **29**, 585-597.
- Texel, P. P. and Williams, C. B. 1997, *Use Cases combined with Booch/OMT/UML: Process and Products* (New Jersey, USA: Prentice Hall).
- Toh, K. T. K., 1999, Modelling architectures: a review of their application in structured methods for information systems application. *International Journal of Production Research* **37**(7), 1439-1458.
- Tonshoff, H. K. and Zwick, M., 1998. An Integrated Product and Process Model. *Book* Globalization of Manufacturing in the Digital Communications Era of the 21st century, (USA: Kluwer Academic Publishers), pp. 209-219.
- Tran, P. and Grewal, S., 1995, Assembly Time Estimation in Assembly Planning Software. *Computer Engineering Division, CED, American Society of Mechanical Engineers* **7**, 47-52.
- Tran, P. and Grewal, S., 1998, A data model for an assembly planning software system. *Computer Integrated Manufacturing Systems* **10**(4), 267-275.
- Van Der Net, A. J.,De Vries, W. A. H.,Delbressine, F. L. M., *et al.*, 1996, A Relation-Based Product Model for Integrating Design and Manufacturing. *Annals of the CIRP* **45**(1), 161-164.
- an Holland, W. and Bronsvoort, W. F., 1996. Assembly Features And Sequence Planning. *Book* Product Modeling for Computer Integrated Design and Manufacture, (Virginia, USA: Chapman&Hall)
- Vernadat, F. B., 1994. Standards and Prenorms in Design, Manufacturing and Automation. *Book* Handbook of design, manufacturing and automation, (New York: Wiley), pp. 993-1019.
- Vos, J. A. W. M., 2000, The Flexfactory project, http://www.wbmt.tudelft.nl/pto/research/flexfactory.html

- Waarts, J. J.,Boneschanscher, N., and Bronsvoort, W. F., 1992, A Semi-Automatic Assembly Sequence Planner. *IEEE international Conference on Robotics and Automation*, , pp. 2431-2438.
- Whitney, D. E., 1996, The Potential for Assembly Modeling in Product Development and Manufacturing, http://web.mit.edu/ctpid/www/Whitney/papers.html
- Williams, T. J., 1998. The Purdue Enterprise Reference Architecture and Methodology (PERA). *Book* Handbook of Life Cycle Engineering, (Dordrecht, The Netherlands: Kluwer Academic Publishers), pp. 289-330.
- Wong, J. H. and Sturges, R. H., 1994, Design for Assembly Factors for Large and Heavy Parts. *Transactions of the ASME* **116**, 508-510.
- Xue, D., Sun, J., Norrie, D.H., 1999, Development of an intelligent system for building product design and manufacturing- Part I: product modelling and Design. *Second International Workshop on Intelligent Manufacturing Systems*, Belgium, , pp. 367-376.
- Yao, Z.,Bradley, H. D., and Maropoulos, P. G., 1998, An aggregate weld Product Model for the early design stages. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM* **12**(5), 447-461.
- Yao, Z.,Bradley, H. D., and Maropoulos, P. G., 1999, Aggregate Process Planning for Sheetmetal Products Part1: product modelling. *15th International Conference on Computer-Aided Production Engineering (CAPE'99)*, Durham, UK, Apr 19-21 1999 , pp. 139-144.
- Yin, C.,Zhong, B., and Yi, H., 1999, Assembly Process Planning and Evaluation for STEP-based integrated design. *Proceedings of the 15th International Conference on Computer-Aided Production Engineering CAPE'99*, , pp. 493-498.
- You, C. F. and Chiu, C. C., 1996, An Automated Assembly Environment in Feature-Based Design. *International Journal of Advanced Manufacturing Technology* **12**, 280-287.
- Young, R. I. M., Bell, R., 1992, Machine Planning in a Product Model Environment. *International Journal of Production Research* **30**(11), 2487-2513.
- Young, R. I. M., Canciglieri, O., Costa, C.A., 1999, Manufacturing information interactions in data model driven design. *Proceedings of the Institution of Mechanical Engineers* **213**(B), 527-532.

- Young, R. I. M. and Bell, R., 1995, Product and Manufacturing Models: Information Tools for the Designer. Loughborough, England., Loughborough University, Department of Manfuacturing Engineering. Loughborough, England.
- Zha, X. F.,Lim, S. Y. E., and Fok, S. C., 1998, Integrated Intelligent Design and Assembly Planning: A survey. *International Journal of Advanced Manufacturing Technology* **14**, 664-685.
- Zha, X. F.,Lim, S. Y. E., and Fok, S. C., 1999a, Development of Expert System for Concurrent Product Design and Planning for Assembly. *International Journal of Advanced Manufacturing Technology* **15**, pp. 153-162.
- Zha, X. F.,Lim, S. Y. E., and Fok, S. C., 1999b, Integrated knowledge-based approach and system for product design for assembly. *International Journal for Computer Integrated Manufacturing* **12**(3), 211-237.
- Zhao, J., Cheung, W. M., Young, R. I. M., 1999, A Consistent Manufacturing Data Model to Support Virtual Enterprises. *International Journal of Agile Management Systems* **1**(3), 150-158.
- Zuobao, W.,Guangleng, X., and Yi, P., 1996, Product Development Process Simulation for Concurrent Engineering. *High Technology Letters* **2**(1), 1-4.

# Appendix A. DESCRIPTION OF METHODS USED IN THE RESEARCH

## A.1. Introduction

Chapter 3 of the thesis presented the research environment in which the research was done. That chapter presented the combination of tools and methodologies applied for defining the structure of information models that support applications through the product development cycle.

Section 3.3.1 explained the object oriented methodology used in the research, which was illustrated in figure 3.7, and provided an introduction to the methods used.

This appendix provides a description of the IDEF0, IDEF3 and UML methods used to support the representation of the information and the development of the experimental system done in this research.

## A.2. IDEF0

IDEF0 is a methodology used for modelling the functions (activities, actions, processes, operations) performed by a system or enterprise, and the functional relationships and data (information or objects) that support the integration of those functions [NIST, 1993 #29].

An IDEF0 model is composed of a hierarchical series of diagrams that display increasing levels of detail describing functions and their interfaces within the context of a system. The two primary modelling components are functions (represented on a diagram by boxes) and the data and objects that interrelate those functions (represented by arrows). The basic representation is shown in Figure A.1.

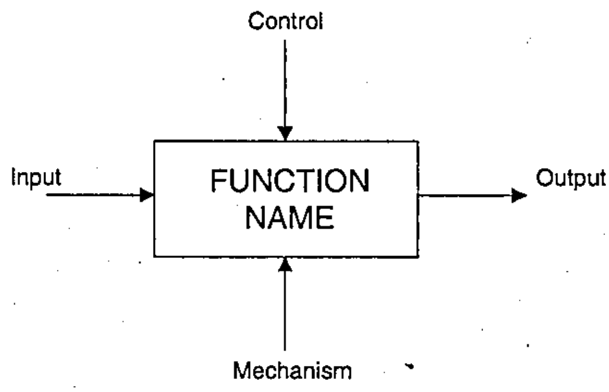



Figure A.1 Basic IDEF0 representation of an activity and its related information

The Function name (inside the box) must be a verb or a verb phrase. The position of the arrows around the box represent different things, the arrows entering the left side of the box are inputs, those are transformed by the function to produce outputs. Arrows entering the box on the top are controls. Controls specify the conditions required for the function to produce correct outputs. Arrows leaving a box from the right side are outputs, the data or objects produced by the function. Arrows connected to the bottom side of the box represent mechanisms. Upward pointing identify some of the means that support the execution of the function. [NIST, 1993 #29]

A function represented in a box can be decomposed in sub-functions by creating a child diagram. In turn, each of these sub-functions can be decomposed, each creating another, lower-level child diagram. Each child diagram must inherit the same arrows entering or leaving the parent function. Figure A.2 shows the decomposition of some of the IDEF0 diagrams created for modelling the concurrent design process. The complete set of diagrams is included in Appendix C.

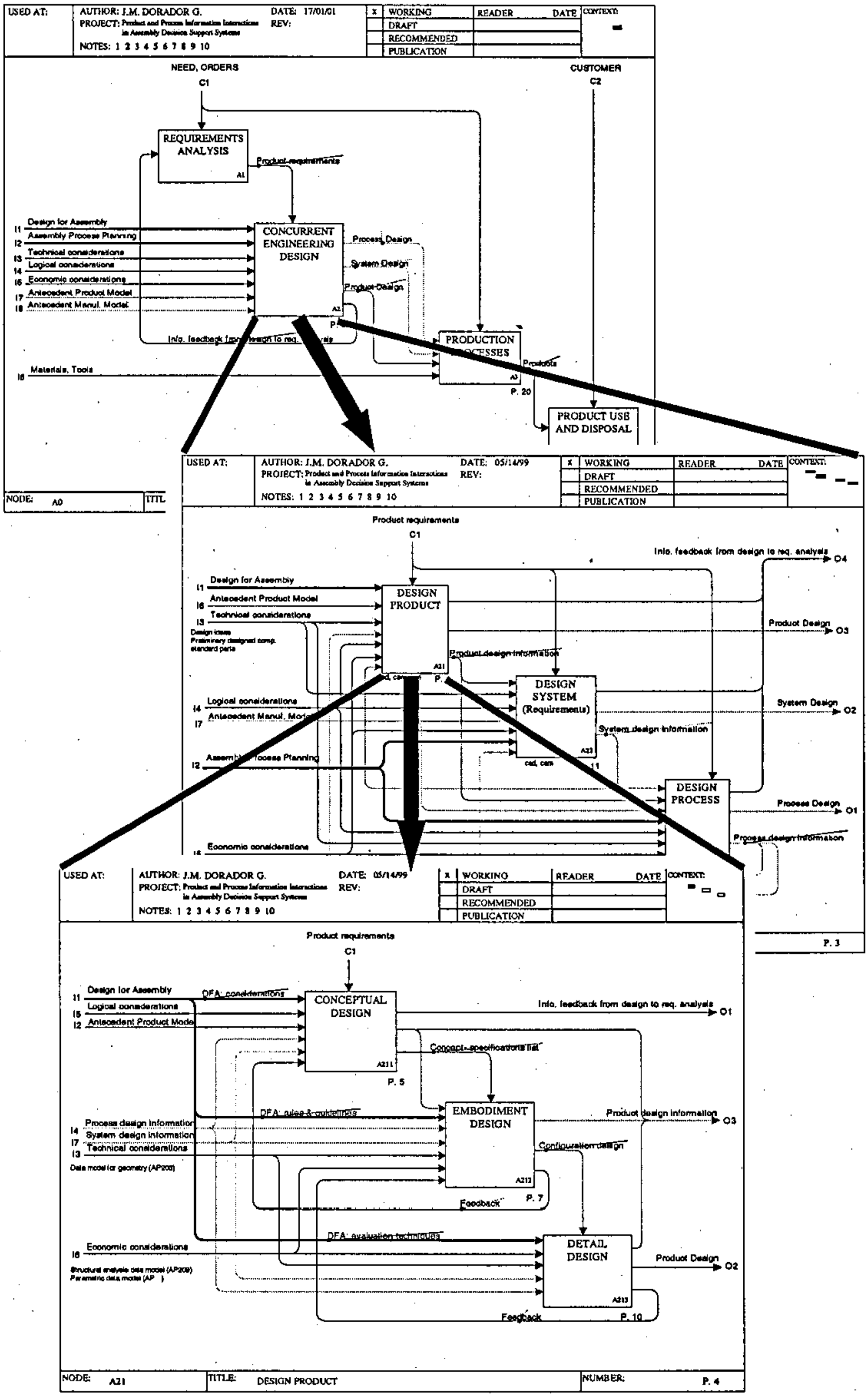



Figure A.2 Decomposition of the IDEF0 diagrams

The IDEF0 methodology can be applied to analyse an existent system, so the result is an *'as is'* model. This methodology can also be applied to obtain an *'as should be'* model [Colquhoun, 1991 #30]. The 'as is' model allows to evaluate the present situation of the system and the 'as should be' helps to define the strategies to follow in order to improve the system by describing the information flow necessary to support each activity.

The first step in applying the IDEF0 methodology is the definition of the context in which the model is done. The diagram at the highest level of an IDEF0 model is known as the 'environment' and is designated a node number A-1. This is used to establish the environment in which the model exists and identifies the inputs, outputs and controls that cross the boundary of the model [Colquhoun, 1991 #30].

Although powerful, IDEF0 has its weaknesses [Ang, 1997 #11]. First, the process of IDEF0 modelling can be time-consuming. This happens even with the aid of software. Second, the process can be inconsistent, due to its subjectivity. The problem occurs because the interfaces of IDEF0 are not precisely defined and are subject to individual interpretations, so it is very likely that different modellers can obtain different IDEF0 models of the same system, despite having the same modelling viewpoint. An important limitation of IDEF0 is its incapability of modelling information process flows, due to the lack of time dependency.

These limitations impose a strong restriction on the use of this model for the design of information systems, so the need for other methodologies to capture the sequence of processes and information structure is evident.

## A.3. IDEF3

The IDEF3 Process Description Capture Method was created specifically to capture descriptions of sequences of activities. Because of this capability, is a good option to use as a method after an IDEF0 model has been created as the activities modelled in IDEF0 can be used as a basis for the definition of the processes. The IDEF3 has been extensively used for modelling processes, aiming to provide a general-purpose description of them. Among the main advantages of this methodology are its simplicity and descriptive power [Huang, 1998 #13], [Kusiak, 1996 #31], [Zuobao, 1996 #37].

The basic structure of IDEF3 is based on a scenario. A scenario can be defined as a set of situations that describe typical class problems addressed by an organisation or a system, or the setting within which a process occurs. Scenarios establish the focus and boundary conditions of a description.

An IDEF3 Process Description is developed using two knowledge acquisition strategies: a *process-centred* schematic and an *object-centred* schematic.

The basic elements of an IDEF3 process-centred diagrams are illustrated in Figure A.3, and explained below using definitions given by Mayer, et al [Mayer, 1995 #17].

In IDEF3 diagrams, boxes represent types of happenings. The neutral term *units of behaviour* (UOB) refer to such happenings. Each UOB box represents a **real-world process.** The information recorded about a UOB includes (1) a name (often verb-based) that indicates what the UOB represents, (2) the names of the objects that participate in the process and their properties, and (3) the relations that hold between the objects.

The arrows (called *links*) connecting the boxes indicate the **precedence relationships** (or more generally *constraints*) that exist between the processes being described. Thus, an instance of the UOB at the source of a link would complete before an instance of the UOB at the end of the same link starts.

The small boxes denote *junctions*. A junction is a point in the process where a process splits into multiple paths, or where multiple paths merge. Junctions represent constraints (or the effects of constraints) of the *activation logic* for the process.

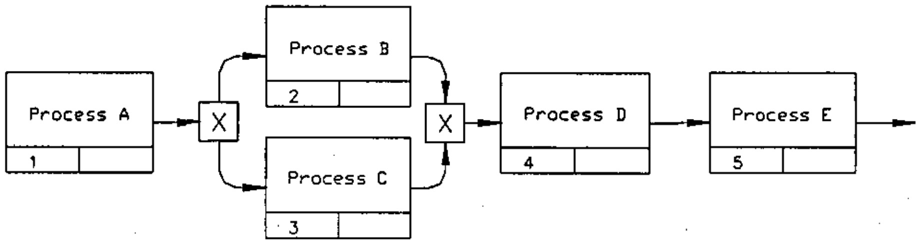



Figure A.3 IDEF3 Unit Of Behaviour boxes linked by relationships and junctions

IDEF3 Process Schematics are the primary means for capturing, managing, and displaying process-centred knowledge. These schematics provide a graphical medium that helps to communicate knowledge about processes. This includes knowledge about events and activities, the objects that participate in those occurrences, and the constraining relations that govern the behaviour of an occurrence.

The purpose of Object Schematic is to identify the possible states in which an object can exist. Though a real-world object often evolves through a continuum of states, an Object Schematic focuses on those distinguished states of particular interest to the domain expert. Figure A.4 shows three states of an object, the boxes that are included between the stages are known as *referents*, and are included to enhance understanding and provide additional meaning, they represent the processes that occur between the stages of the analysed object. They are also a link between the process and object schematics.

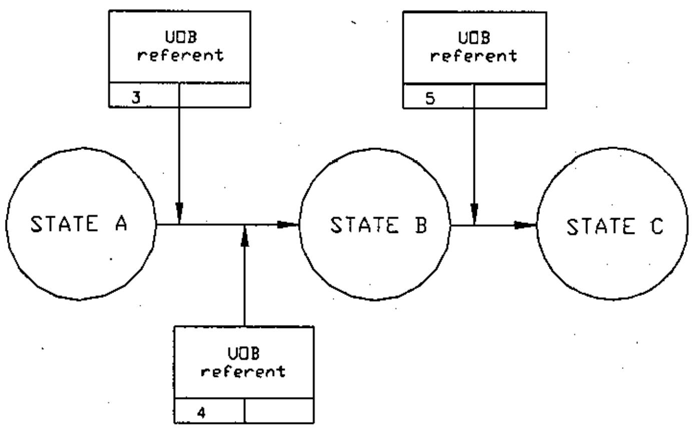



Figure A.4 Basic elements in an Object-centred description

IDEF3 also includes a set of Elaboration Forms to aid the process of capturing information. In these forms the name of the UOB is captured together with the objects involved in the process, facts, constraints and description. An example of these Elaboration Forms is shown as Figure A.5.

| | | | | | |
|---|---|---|---|---|---|
| USED AT<br>Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 1999<br>PROJECT Product and Process Information Interactions in Assembly Decision Support Systems<br>NOTES 1 2 3 4 5 6 7 8 9 REV: | X | WORKING<br>DRAFT<br>RECOMMENDED<br>RELEASED | REVIEWER | DATE: |

| | |
|---|---|
| UOB No. 61 | **UOB Name:** GET PROCESS INFORMATION<br>**Objects:** Information from the process<br>**Facts:** The information comes from activity No. 30<br>This information is used in combination with the information received in activity 60<br>**Constraints:** This information must arrive as soon as it is created or updated.<br>**Description:** The information received contains the result of the analysis of operations done in the detail stage of the design of theprocess. |
| UOB No. 62 | **UOB Name:** SELECTION OF ELEMENTS FOR MANIPULATION<br>**Objects:** Information from the design of the product and the system<br>Information about resources available for manipulating the parts<br>Designer<br>**Facts:** The designer must select which elements will be necessary for performing the manipulation of the parts.<br>The design of the parts must be considered in the selection of the elements.<br>**Constraints:** The manipulation can be manual or automatic, according to economic parameters already studied.<br>The elements selected must be able to grasp, orient and insert the parts in the appropriate position in the assembly<br>**Description:** The selection of elements for manipulation must consider the design of the product and the processes that must be performed for the assembly. Also the interaction with other elements selected for performing the assembly must be considered. |

| | | |
|---|---|---|
| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE<br>UOB Elaboration: |

Figure A.5 Example of an IDEF3 Elaboration form.

The notation and format in which IDEF3 models present the information is very useful for communicating with collaborators or future users of the system under development. On the other hand, the format is still very descriptive and unstructured. It is therefore inadequate as a basis for building the software structure. So, other methods have to be used to identify the classes with their attributes and behaviour, which constitute the object-oriented structure of the software.

The IDEF3 diagrams elaborated during the application of the methodology explained in chapter 3, are included in Appendix D.

## A.4. UML

The Unified Modelling Language is the result of the merging of three modelling methods, Booch, OMT and Objectory, taking some concepts from other methods. UML is a language used to specify, visualise and document the artefacts of an object-oriented system under development [Quatrany, 1998 #22].

The Unified Modelling Language (UML) is a standard modelling language that must be applied in the context of a process. The UML authors propose a development process named Rational Unified Process, explained in [Quatrany, 1998 #22]. Texel and Williams [Texel, 1997 #23] propose other approach, which is more detailed in the steps that should be followed in the creation of a system with Use Cases.

The elements that are used in the construction of the models are:

- Structural elements: Class, interface, collaboration, use case, active class, component, node
- Behavioural elements: Interaction, state machine
- Grouping elements: Package, subsystem

The relationships between elements can be dependencies, associations, generalisations, and realisations.

In terms of the views of a model, UML defines several diagrams, including Use Case Diagrams for capturing the system functionality, Class Diagrams, and various behaviour diagrams for capturing the dynamic behaviour. These diagrams provide multiple perspectives of the system under analysis or development. The underlying model integrates these perspectives so that a self-consistent system can be analysed and built.

The Rational Unified Method is an iterative and incremental process that evolves continuously into the final system. The iterations consist of one or more of the following process components: requirements capture (what the system should do), analysis and design (how the system will be realised), implementation (production of the code), and test (verification of the system). All of these analysed through the following phases: inception (specifying the project vision), elaboration (planning the activities), construction and transition (supplying the product to the user).

UML captures and documents decisions made during the inception and elaboration phases of system development [Quatrany, 1998 #22]. In the inception phase, UML uses Use Case Diagrams, in which the relationships between actors and use cases are represented. The actors represent anyone or anything that must interact with the system. Use Cases represent the functionality of the system, there are two kind of relationships between them, *uses* and *extends*. The basic elements used in the Use Cases diagrams are represented in Figure A.6.

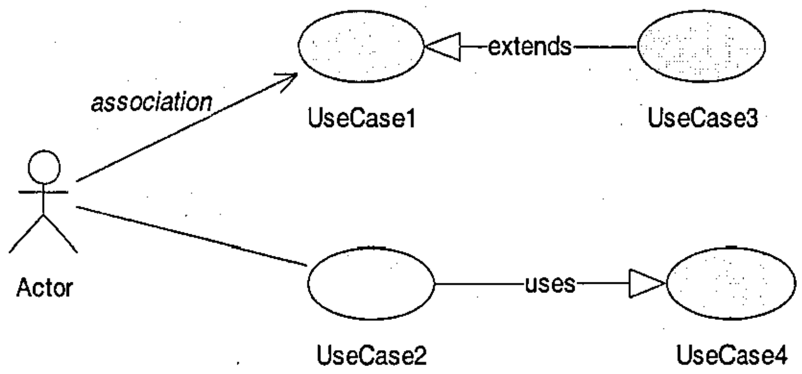



Figure A.6 Basic elements in a UML Use Case representation

In the elaboration and construction phases, the Class Diagrams and State Transition diagrams are used. A class is a description of a group of objects with common properties, common behaviour, common relationships with other objects, and common semantics. The classes are represented as rectangles in which the name of the class is defined together with their attributes and operations. The relationships with other classes are represented using lines with different endings signifying associations, aggregation, or other relationships, as shown in Figure A.7.

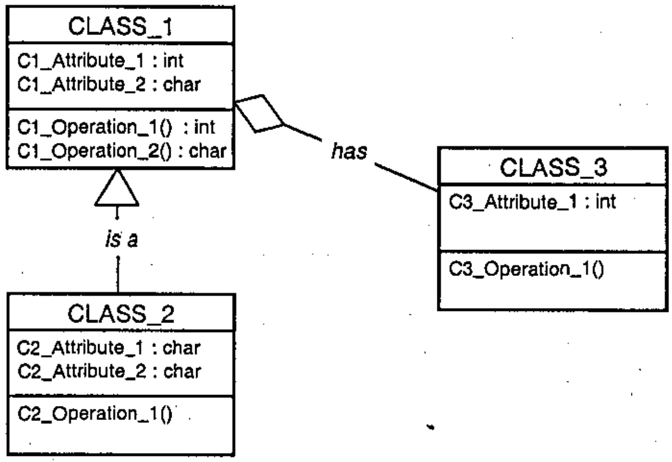



Figure A.7 Basic elements in a class diagram

In the work described here, where the definition of information models is critical, the most important aspects of UML are the class diagrams. These can be generated either from the application of Use Cases or from an evaluation of the IDEF3 models. In software systems development the full rational unified process could be used on its own. However the models generated need a higher level of expertise to be able to understand them, which makes communication with potential system users significantly more difficult. The initial use of IDEF0 and IDEF3 diagrams are a better way for communicating with collaborating companies than using Use Case diagrams.

# Appendix B. Paper published in the IJCIM

. J. COMPUTER INTEGRATED MANUFACTURING, 2000, VOL. 13, NO. 5, 430–445

# ɩpplication of IDEF0, IDEF3 and UML ıethodologies in the creation of information ıodels

J. M. DORADOR and R. I. M. YOUNG

ıstract. The importance of information models is widely cognised by the computer integrated manufacturing search community. However, improved methods are still eded to assist the developer in the definition of information odel structures. Currently available methods and standards n only help in certain stages of the information modelling ocess. This paper explores the benefits of combining three ethodologies in the definition of the structure of informa- ɔn models that support applications through the product life cle. While all the methodologies that are currently used in formation modelling have some advantages, they also have eaknesses. This paper shows how the combined use of IDEF0, )EF3 and UML methodologies can be used to advantage in .e context of the Open Distributed Processing (ODP) andard ISO 10746. The combination of these methods means .at the weaknesses of each can be counteracted by the rengths of others. IDEF0 is commonly used, with some ıccess, to model enterprise activities and information flows. .owever its ability to describe information is weak and it annot model process flows. IDEF3 offers both a process flow apability that can be linked to IDEF0 and an ability to model ıformation in object centred descriptions. The results of the pplication of these methodologies provide the required ıformation for the top levels of the RM-ODP. Furthermore, oth methodologies have been proved to be very good means f communication with the collaborating companies. The esults obtained with the IDEF3 method provide key ıformation for the creation of classes with attributes and perations that can then be used in the design of computa- ional systems using UML. The resulting UML class diagrams how the relationships and inheritances that are the main nput to the creation of object-oriented databases that hold the lata of the information models. These combined methods ıave been used to model the information required through lifferent stages of the product life cycle for the assembly of arge electrical machines and have been shown to provide an mproved definition of the relationships between the stages of nformation model definition.

*Authors:* J.M. Dorador and R.I.M. Young, Department of Manufacturing Engineering, Loughborough University, Loughborough, Leics., LE11 3TU, UK.

## 1. Introduction

Increasingly competitive markets demand rapid responses from industry, providing better products faster with reduced prices. In order to fulfil those requirements there have been many improvements and developments in materials, processes, tools for the designer, quality improvement, and integration of the enterprise's activities.

One of the main characteristics of concurrent engineering is the intensive information interchange in the early stages of product design. This includes sharing information about products and the processes and resources required in their production. Different areas of the enterprise, using a variety of computer software applications, use the same information, so it is convenient to have it centrally stored in information models. The issue of information modelling has been widely recognised and studied by the international research community.

Information models can be divided into two different models, namely the product model and the manufacturing model. The first supports applications used in product development and the latter contains information of the manufacturing facilities available to manufacture the product. A product model is a computer representation of product data that contains detailed information about a product or a family of products, so it can support the applications that are interacting in the product's life cycle. The product model must allow those applications to have access to the information and also to be a repository for the data created by them. Therefore, the product model must be structured in such a way that the applications know where to find and to store information (Baxter 1994, MOSES 1992, Krause *et al.* 1993). The term manufac-

*International Journal of Computer Integrated Manufacturing*
ISSN 0951-192X print/ ISSN 1362-3052 online © 2000 Taylor & Francis Ltd
http:/ / www.tandf.co.uk/ journals

uring model is used to define, describe and capture the nanufacturing situation of a particular enterprise Molina 1995, Young and Bell 1995). The main objective of the manufacture model is to provide lesigners and manufacturing engineers with high quality manufacturing information on which to base heir decisions.

STEP (Standard for the Exchange of Product Model Data), ISO standard 10303, defines a neutral data ormat for the representation and exchange of product lata. The goal of this standard is to complete a system-ndependent representation of all product-related data luring the product life cycle (Krause *et al.* 1993, Ashworth *et al.* 1996). The technical committee ISO TC184/ SC4 is working on the development of a tandard structure for a product model (NIST 1997). The methods that are commonly used in the generaion of STEP information models are IDEF0 and the Express language.

In order to capture the requirements that must be contained in information models, several approaches have been proposed based on reference architectures. Some of them have become standards, like CIM-OSA Computer Integrated Manufacturing Open System Architecture) (Jorysz and Vernadat 1990) and RM-ODP (Reference Model for Open Distributed Processing) (Toh 1999). Other important methods are the Grai Integrated Methodology (Doumeingts 1998) and he Purdue Enterprise Reference Architecture (PERA) Williams 1998).

CIM-OSA focuses on modelling the complete nterprise from requirement and specification through ystem design, implementation, operation and maintenance. RM-ODP is used for software systems development. They can provide a clear picture of the organisation and assist in the definition of the nformation needs. They can also be used to evaluate he impact of decisions in the different areas of the nterprise (Prasad 1996).

RM-ODP provides a framework for designing coherent flexible distributed systems captured in the SO 10746 standard. The architecture of the RM-ODP provides a standard upon which a wide range of pecifications for distributed systems can be integrated Toh 1999). Because of the complexity of modelling arge amounts of information and capturing all aspects of information, the RM-ODP is divided into five iewpoints that express concepts and rules relevant to particular area in terms of which a system can be escribed. This standard is used as a reference in this work.

Within RM-ODP the general functions that the nformation system will perform are defined in the nterprise viewpoint. At the information viewpoint, the system is described in terms of information structures and flows. The other three viewpoints are for software implementation support, in the computational level, the system is described as a set of objects that interact at interfaces. In the engineering viewpoint, the mechanisms that support the system distribution are defined; and at the technology viewpoint, the components from which the system is constructed are defined (McKay *et al.* 1996).

## 2. Methodologies applied in the definition of the information models

In figure 1 the cylinders inside the product model and the manufacturing model represent the information structure related to assembly, defined in order to support the Assembly-related activities during the product life cycle. In the lower level of the diagram, the DFA and the APP applications are illustrated with an arrow between them representing the interaction that is being explored. This interaction will allow the designer to take into account the main APP considerations when applying the DFA during the design stages of the product life cycle. Therefore, only some details will have to be defined in the post-design stages of the product life cycle.

In this research, the experimental information models developed to test the ideas was designed using the RM-ODP structure, as shown in figure 2. The enterprise view level of the information models was developed using IDEF0 and IDEF3 process centred descriptions. This enables the enterprise activities and process flows to be modelled with some indication of information flows. In the information view level, the IDEF3 object centred descriptions and elaboration diagrams were used to document the process and provide a view of the information objects needed and their states. UML Class diagrams were then used to construct the computational view of the system. The experimental software system was elaborated using object-oriented technology.

The use of each of these methods and the value of using them in combination is explained in the next section.

## 3. IDEF0

IDEF0 provides a means for modelling the functions (activities, actions, processes, operations) required by a system or enterprise, and the functional relationships and data (information or objects) that support the integration of those functions.

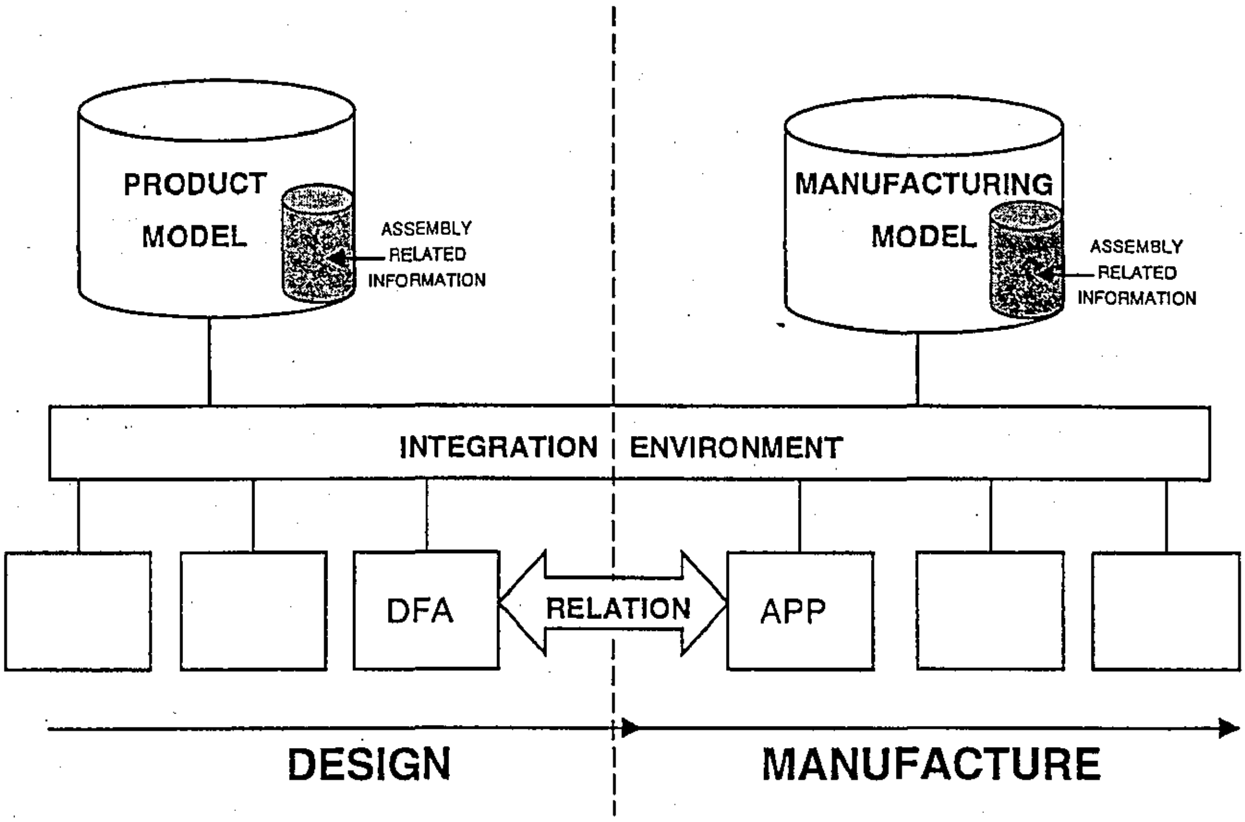



Figure 1. Information models and assembly application interactions (Dorador and Young 1999).

An IDEF0 model is composed by a hierarchical eries of diagrams that display increasing levels of detail escribing functions and their interfaces within the ontext of a system (NIST 1993). The two primary iodelling components are functions (represented on a iagram by boxes) and the data and objects that inter-elate those functions (represented by arrows), as hown in figure 3.

A set of IDEF0 diagrams can exist as either generic is is' or 'as should be' models (Colquhoun and Baines 991). The 'as is' model allows to evaluate the present ituation of the system and the 'as should be' helps to efine the strategies to follow in order to improve the ystem by describing the information flow necessary to upport each activity.

The use of IDEF0 is widely used in the research ommunity due to its flexibility and clarity for model-ing activities and the information flows between them. : is particularly good for evaluating enterprise activity ructures, as it provides an easy to understand model hich non-experts can understand and assess. Although DEF0 is good at providing an initial view of activity ecomposition, it is incapable of modelling information rocess flows. This is due to its lack of time dependency iput. The representation of information in IDEF0 is also weak as it is captured purely by simple textual description. These limitations impose a strong restriction on the use of this model for the design of information systems, so the need for other methodology able to capture the sequences of processes and information structure is evident.

## 4. IDEF 3

The IDEF3 Process Description Capture Method was created specifically to capture descriptions of sequences of activities. Because of this capability, it is a good option to use as a method after an IDEF0 model has been created as the activities modelled in IDEF0 can be used as a basis for the definition of the processes. The IDEF3 has been extensively used for modelling processes, aiming to provide a general-purpose description of them. Among the main advantages of this methodology are its simplicity and descriptive power (Huang and Kusiak 1998, Kusiak and Zakarian 1996, Zuobao *et al.* 1996).

An IDEF3 Process Description is developed using two knowledge acquisition strategies: a process-centred schematic and an object-centred schematic.

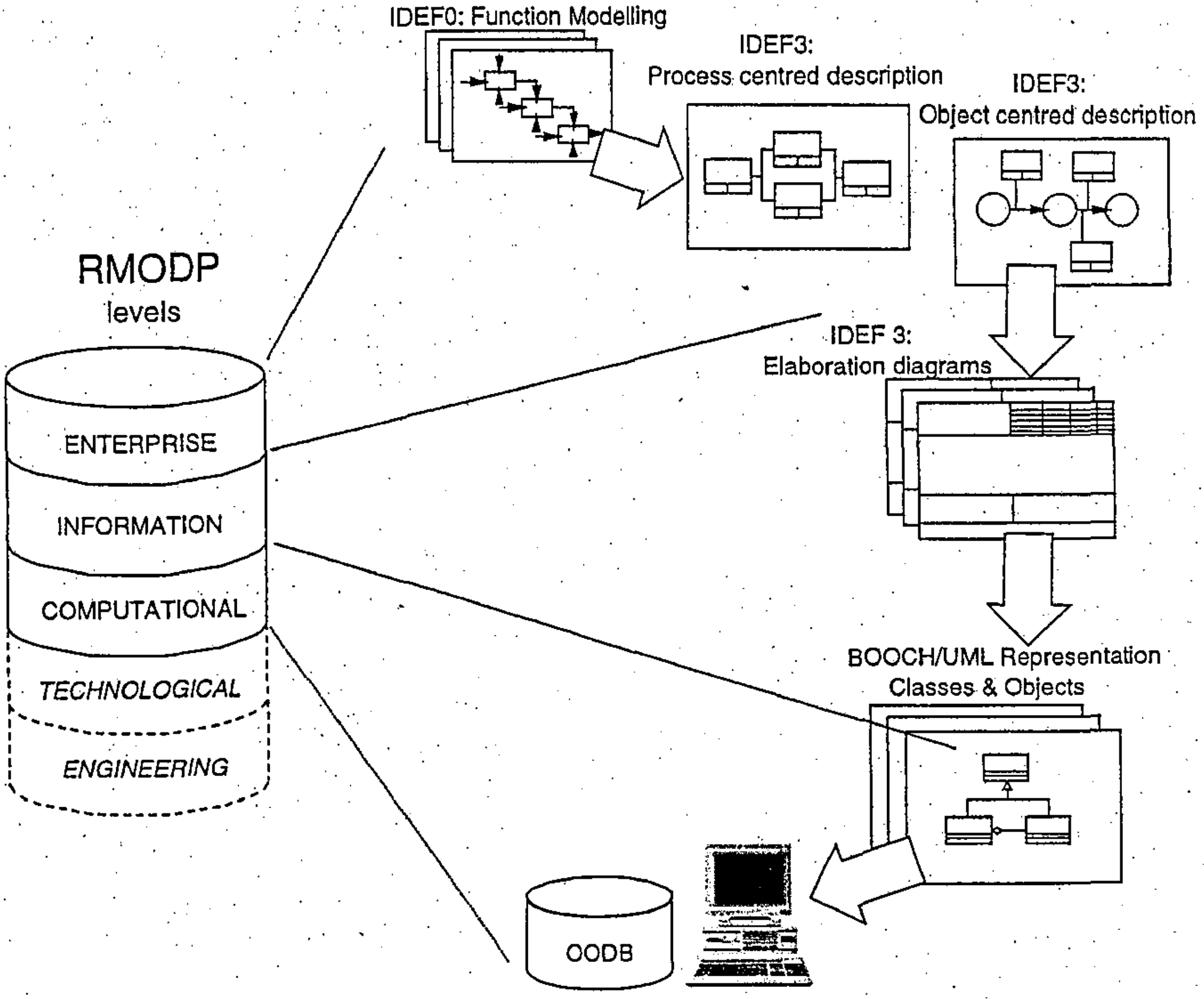
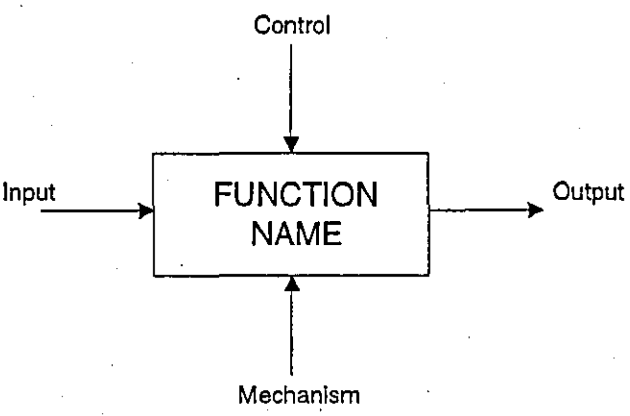



Figure 2. Methodologies that will be used to create the information systems (Dorador and Young 1999).




igure 3. Basic IDEF0 representation of an activity and its related information.

In IDEF3 process-centred diagrams, boxes represent types of happening, as depicted in figure 4. The term units of behaviour (UOB) refers to such happenings. Each UOB represents a real-world process. The arrows connecting the boxes indicate the precedence relationships that exist between the processes being described. Junctions represent constraints of the activation logic for the process. The junctions are used to represent different conditions in which an activity is carried out, as if they were logical operators: and, or, and exclusive or (Mayer *et al.* 1995).

The purpose of the object schematic is to identify the possible states in which an object can exist. Though a real-world object often evolves through a continuum of states, an object schematic focuses on those distinguished states of particular interest to the domain xpert. In figure 5 three states of an object are shown; he boxes that are included between the stages are nown as referents, and are included to enhance nderstanding and provide additional meaning. They epresent the processes that occur between the stages of he analysed object. They are also a link between the rocess and object schematics.

IDEF3 also includes a set of Elaboration Forms to aid he process of capturing information. In these forms the ame of the UOB is captured together with the objects volved in the process, facts, constraints and description.

The notation and format in which IDEF3 models resent the information is very useful for communicatng with collaborators or future users of the system nder development. On the other hand, the format is till very descriptive and unstructured. It is therefore nadequate as a basis for building the software structure. So other methods have to be used to identify the classes with their attributes and behaviour, which constitute the object-oriented structure of the software.

## 5. UML methodology

Unified Modelling Language is the result of the merging of three modelling methods, Booch, OMT and Objectory, taking some concepts from other methods. UML is a language used to specify, visualise and document the artefacts of an object-oriented system under development (Quatrany 1998).

The Unified Modelling Language (UML) is a standard modelling language that must be applied in the context of a process. The UML authors propose a development process named the Rational Unified

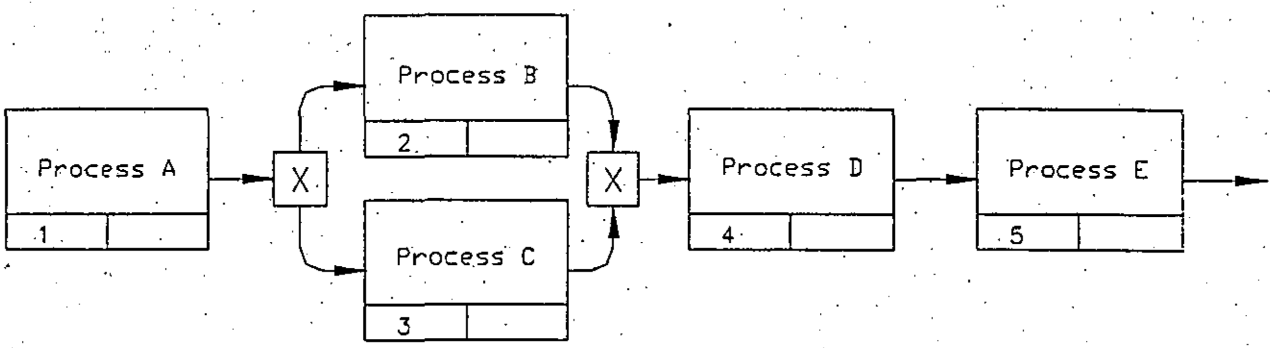



Figure 4. IDEF3 unit of behaviour boxes linked by relationships and junctions.

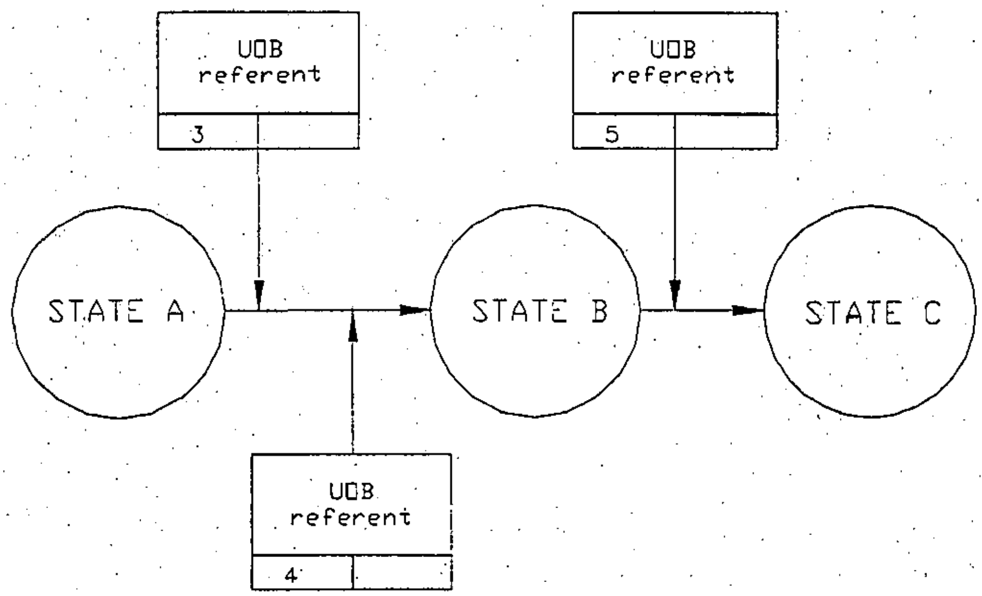



Figure 5. IDEF3 Object schematic description, showing an object in three stages with unit of behaviour referent boxes.

'rocess, explained by Quatrany (1998). Texel and Villiams (1997) propose other approach which is more etailed in the steps that should be followed in the reation of a system with use cases.

The elements that are used in the construction of he models are:

(a) structural elements: class, interface, collabora tion, use case, active class, component, node;
(b) behavioural elements: interaction, state machine;
(c) grouping elements: package, subsystem.

'he relationships between elements can be dependencies, ssociations, generalisations and realisations.

In terms of the views of a model, UML defines everal diagrams, including use case diagrams for apturing the system functionality, class diagrams, and arious behaviour diagrams for capturing the dynamic ehaviour. These diagrams provide multiple perspecives of the system under analysis or development. The nderlying model integrates these perspectives so that a elf-consistent system can be analysed and built.

The rational unified method is an iterative and ncremental process that evolves continuously into the nal system. The iterations consist of one or more of ne following process components: requirements capure (what the system should do), analysis and design how the system will be realised), implementation production of the code), and test (verification of the ystem). All of these are analysed through the following hases: inception (specifying the project vision), laboration (planning the activities), construction and ransition (supplying the product to the user).

UML captures and documents decisions made uring the inception and elaboration phases of system evelopment (Quatrany 1998). In the inception phase, JML uses use case diagrams, in which the relationships etween actors and use cases are represented. The ctors represent anyone or anything that must interact ith the system. Use cases represent the functionality of ne system, there are two kind of relationships between nem, uses and extends.

In the elaboration and construction phases, the class iagrams and state transition diagrams are used. A class is description of a group of objects with common roperties, common behaviour, common relationships ith other objects, and common semantics. The classes re represented as rectangles in which the name of the lass is defined together with the attributes and operaons. The relationships with other classes are represented sing lines with adornments signifying associations, ggregation, or other relationships, as shown in figure 6.

The state transition diagrams represent the states of single object, the events or messages that cause a transition from one state to another, and the actions that result from a state change.

In the work described here, where the definition of information models is critical, the important aspect of UML is the class diagrams. These can be generated from an evaluation of the IDEF3 models. In software systems development the full rational unified process could be used on its own. However the models generated need a higher level of expertise to be able to understand them, which makes communication with potential system users significantly more difficult. The initial use of IDEF0 and IDEF3 diagrams are a better way for communicating with collaborating companies than using use case diagrams.

## 6. Structure of the information models

In order to define the structure for the information models, there are basically two approaches that can be followed. The first is a top-down approach, which starts with the analysis of the enterprise level in order to obtain information about the processes, operations and objects that are involved in the process. The following steps are the definition of the information and computational levels. The other methodology is a bottom-up approach, in which the process starts with the definition of the functionality of the software, and from that point the information and enterprise levels are analysed.

In this research a combination of both methodologies was followed; the top-down approach was applied in order to determine the enterprise and information levels, that is, the environment and external conditions that are necessary for the development of the system. Once that information is known, the bottom-up strategy is applied using UML to complete the basic structure of the information models.

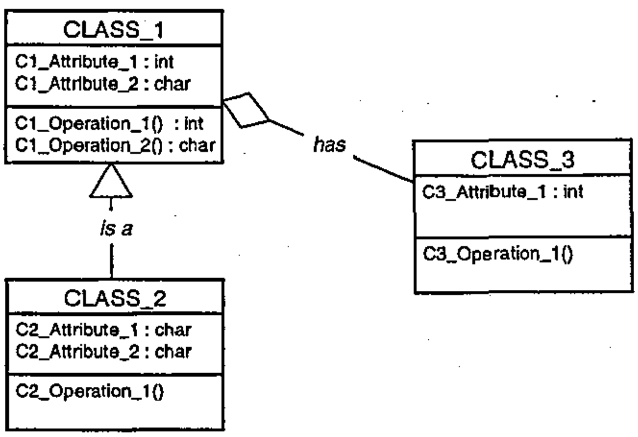



Figure 6. Basic structure of the class diagrams.

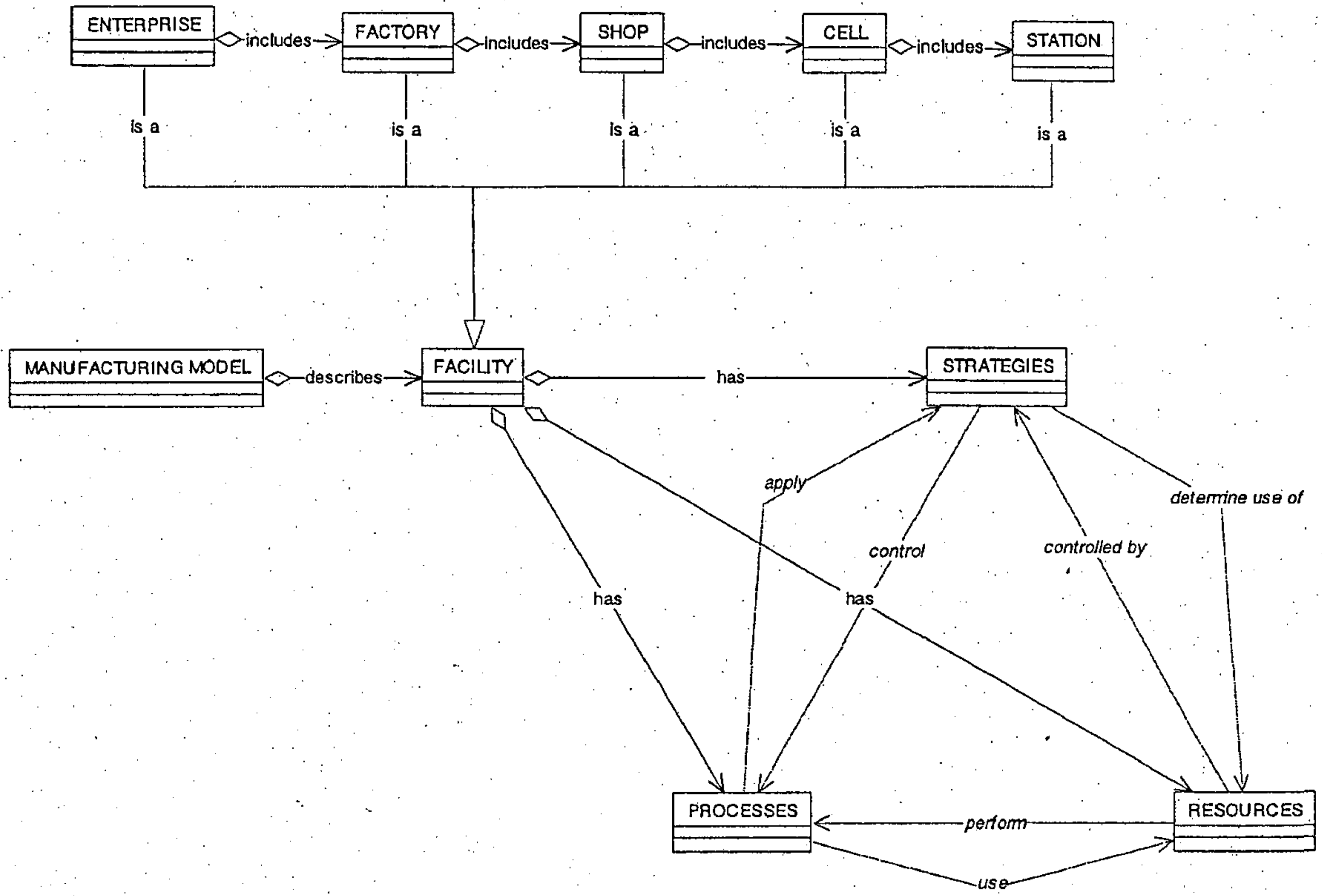



Figure 7. Basic structure of the manufacturing model using UML notation.

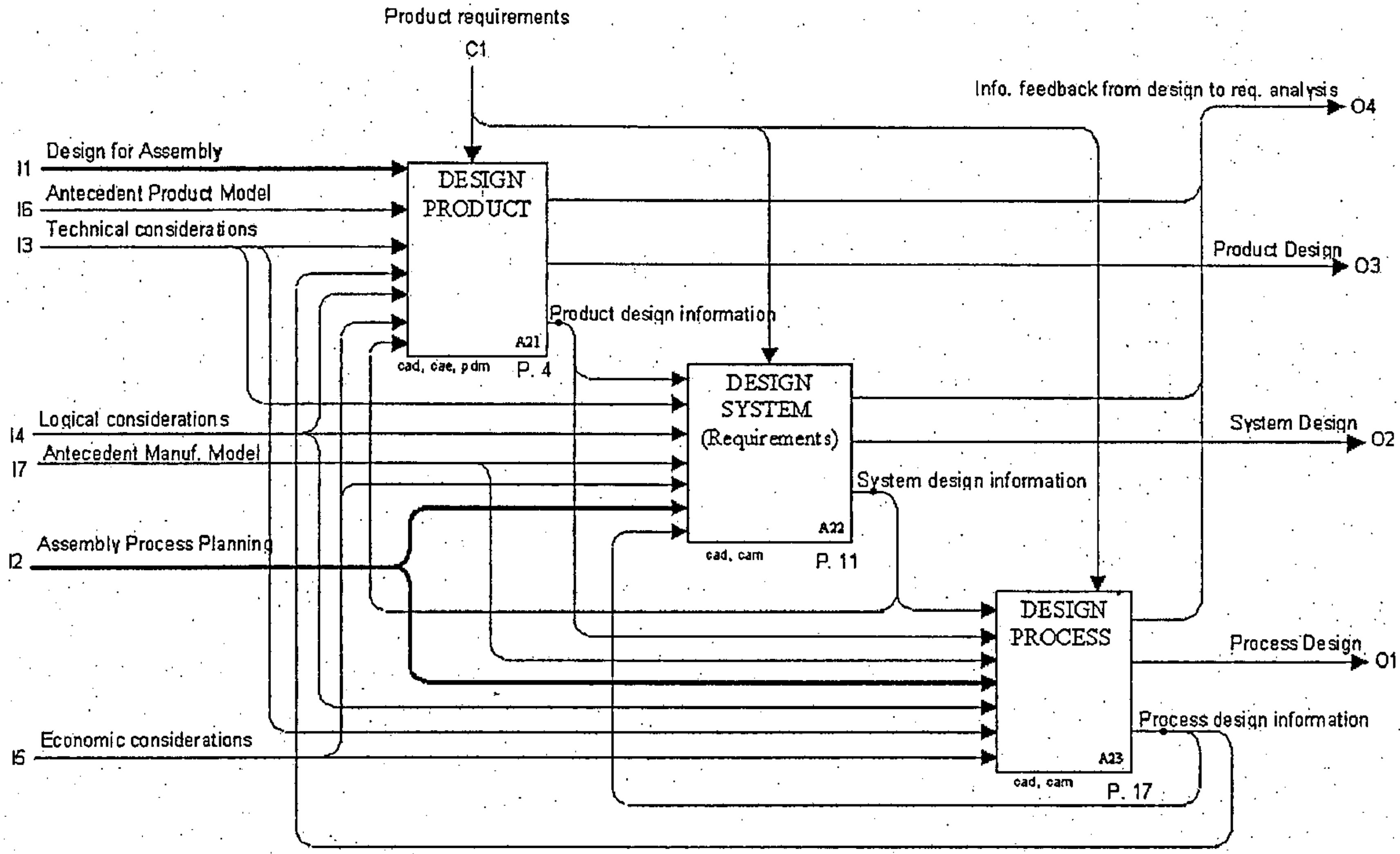



Figure 8. IDEF0 diagram for the concurrent design of the product, system and process.

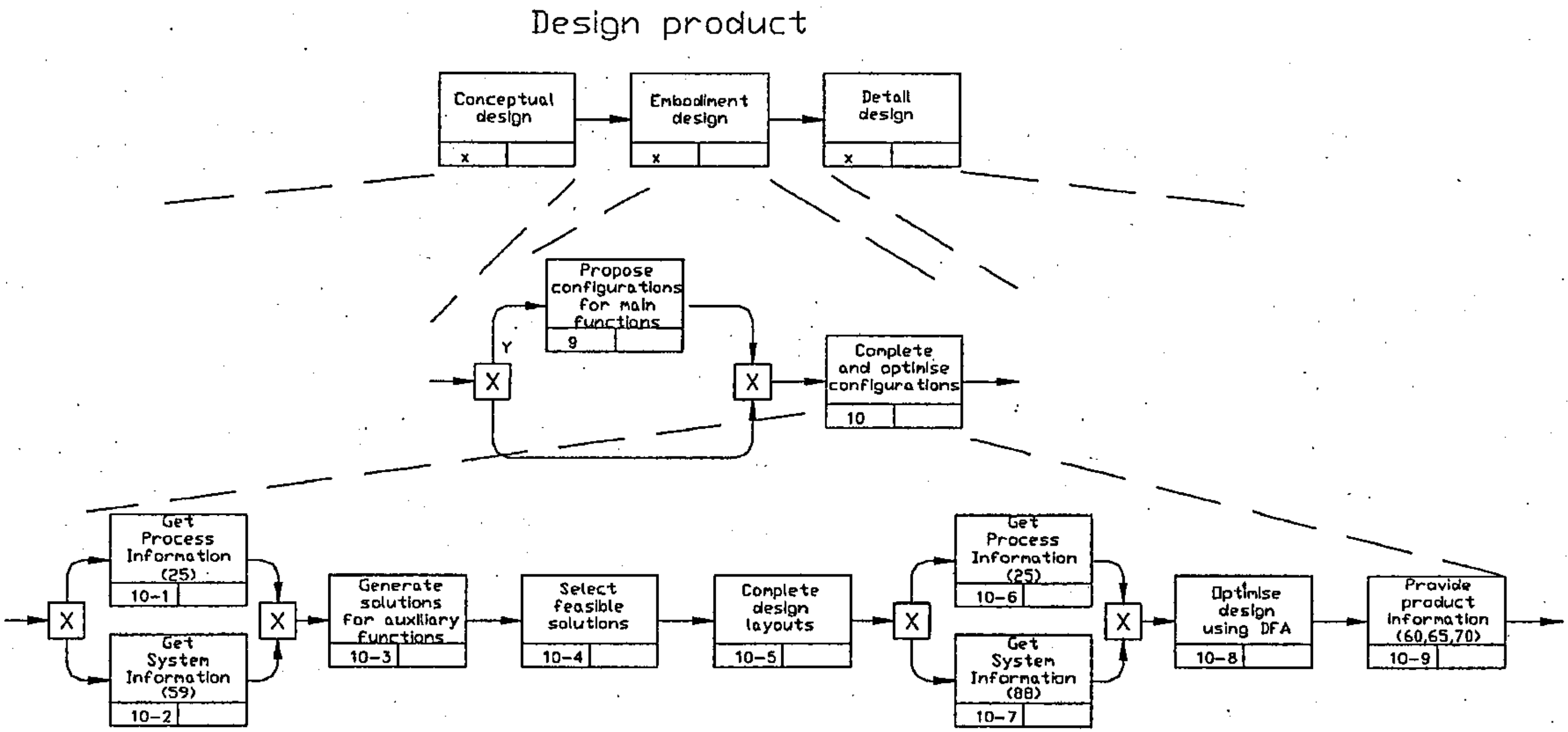



Figure 9. Part of the IDEF3 model for design product.

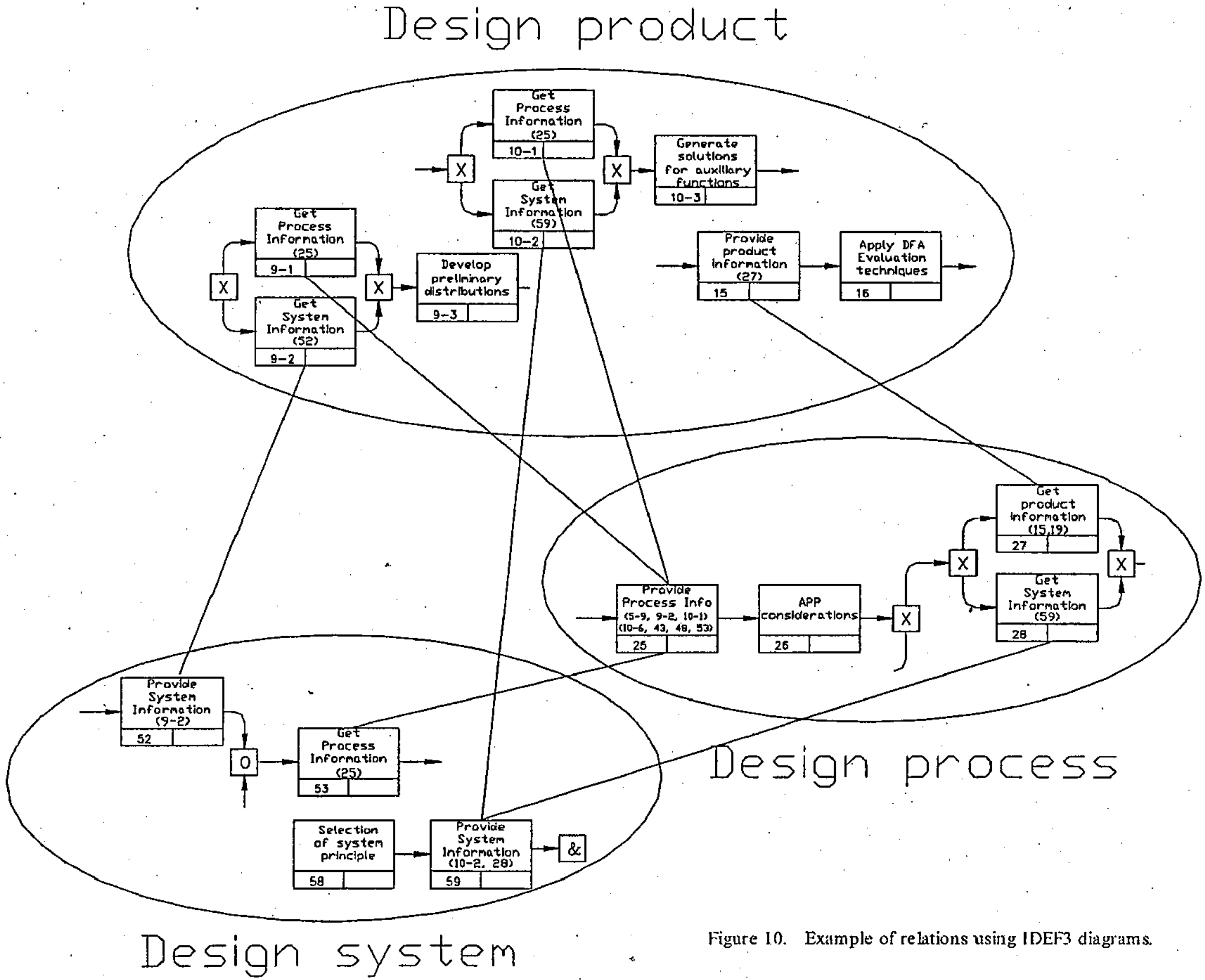



Figure 10. Example of relations using IDEF3 diagrams.

## . Basic structure of the manufacturing model

The basic entities in the definition of a manufactur- ng environment are resources, processes and strate- ies. The resources are the physical elements that nable the product realisation. The processes can be nformational or material and explain the use of the esources. The representation of resources and pro- esses provides a consistent representation of the nanufacturing facilities and their capabilities. In addi- ion, the strategies are represented as the way in which he decisions are made on the use and organisation of esources and processes.

The basic structure for the manufacturing model is resented in figure 7. It is divided in functional levels in rder to allow the capture of the manufacturing unction and behaviour. These levels are enterprise, actory, shop, cell, and station level. This structure is dapted from the MOSES project, developed at oughborough and Leeds Universities (Young and Bell 995, Molina 1995). The enterprise class was included o give the manufacturing model the possibility of apturing information of more than one factory, as equired in the modelling of global enterprises.

## . Application of the methods to define the detailed structure of the information models

In the top-down approach the IDEF0 and IDEF3 nethodologies were applied to produce models that elp to get knowledge about the processes, operations nd objects that are involved in the process, and need o be supported by data from the information models.

Once the basic functionality has been established, the following step is to define the basic structure of the information models. The requirements that were identified in the IDEF models are included in the basic structure.

The IDEF0 model for assembly interactions during the product life cycle was created considering that the design of the product, process and system were performed concurrently, as shown in figure 8. The complete product life cycle, from the analysis of the customer requirements through to the shipping of the product was modelled using IDEF0, emphasising the assembly-related activities. The product, system and process designs were divided in conceptual design, embodiment design and detail design stages, in order to analyse the interactions between them in a concurrent engineering environment and their relation with design for assembly and assembly process planning.

Once the phase of building the IDEF0 model for capturing the activities was completed, the IDEF3 methodology was used to capture the process flows, which cannot be captured in IDEF0. The use of IDEF3 methodology does not require a previous modelling in IDEF0, but when the system that is analysed is complex, it is convenient to perform a preliminary modelling in IDEF0. Due to the fact that in this research the assembly interactions through the product life cycle in a concurrent environment are studied, it was decided to perform both the IDEF0 for the activity modelling and the IDEF3 for the process and object centred descriptions.

Three separate IDEF3 models were created; one for design product, one for design system and other for design process, including the interactions with DFA and

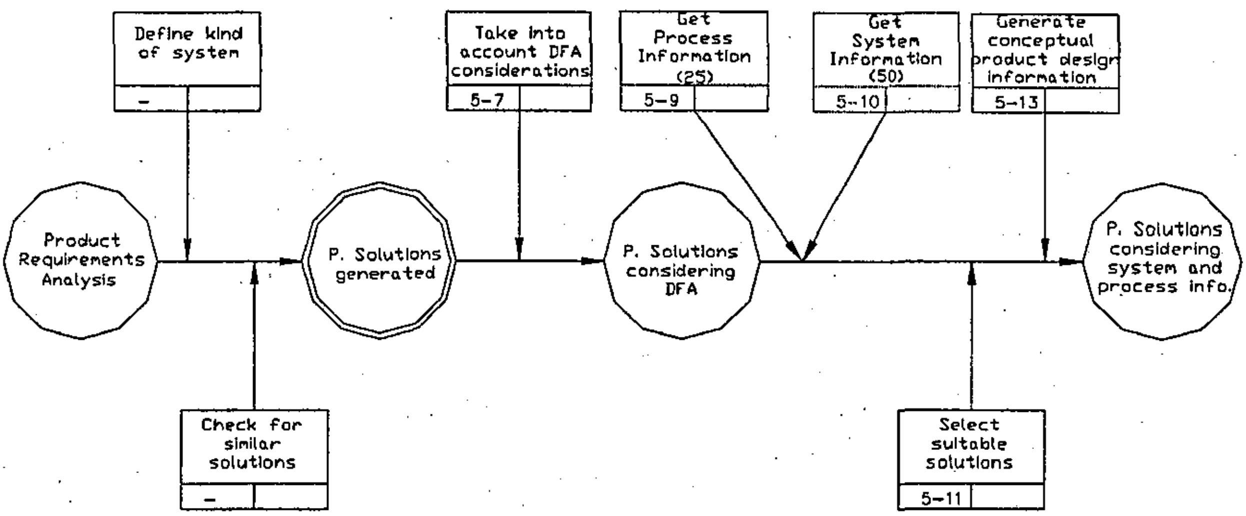
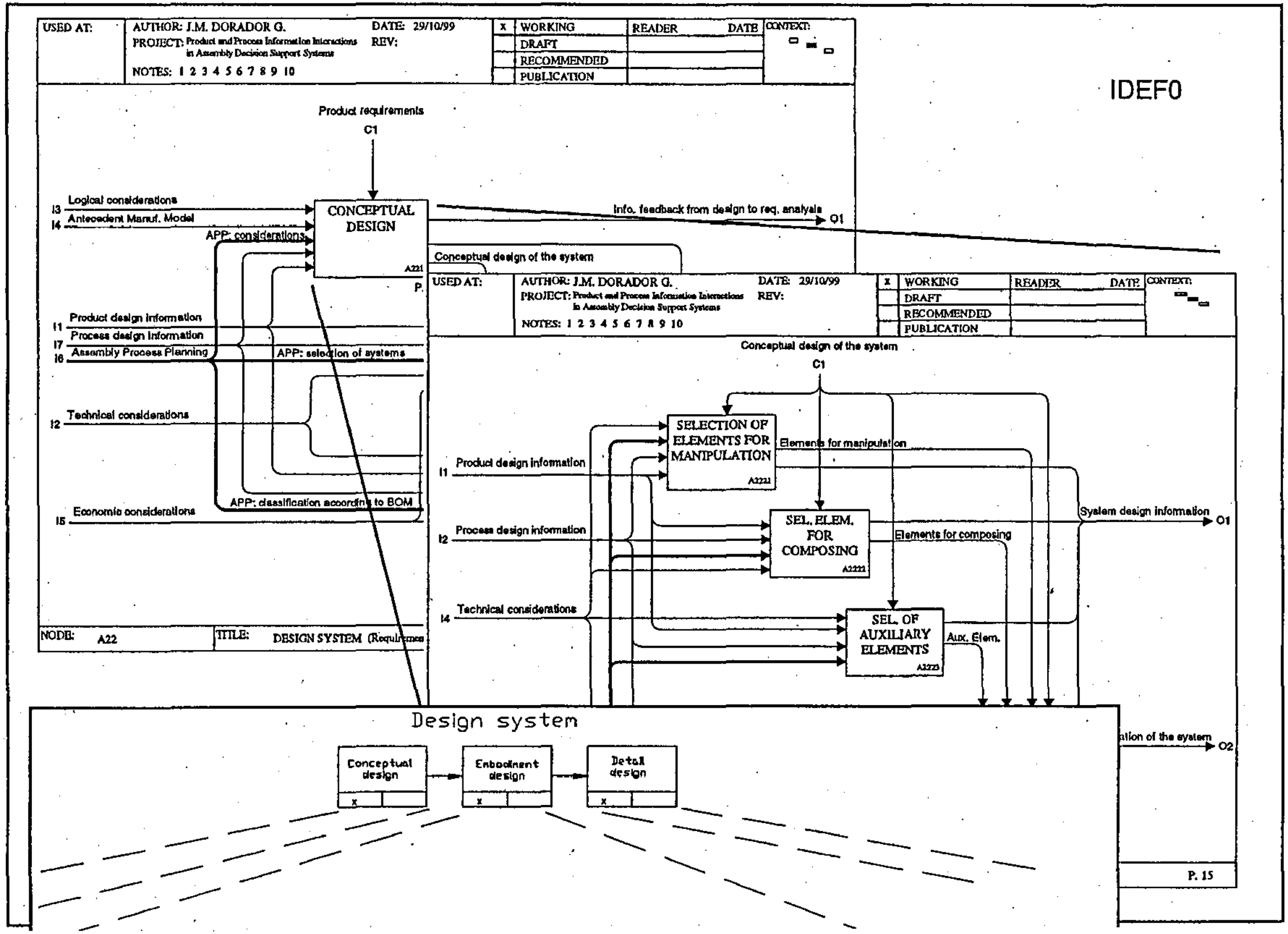



Figure 11. Extract of the object-centred description for the product design.


IDEF0
USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 29/10/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
I3 Logical considerations
I4 Antecedent Manuf. Model
APP: considerations
CONCEPTUAL DESIGN
A221
Info. feedback from design to req. analysis
O1
Conceptual design of the system
I1 Product design information
I7 Process design information
I6 Assembly Process Planning
APP: selection of systems
I2 Technical considerations
APP: classification according to BOM
I5 Economic considerations
NODE: A22
TITLE: DESIGN SYSTEM (Requirements
USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 29/10/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Conceptual design of the system
C1
SELECTION OF ELEMENTS FOR MANIPULATION
A2221
Elements for manipulation
I1 Product design information
SEL. ELEM. FOR COMPOSING
A2222
System design information
O1
I2 Process design information
Elements for composing
I4 Technical considerations
SEL. OF AUXILIARY ELEMENTS
A2223
Aux. Elem.
...ation of the system
O2
P. 15
Design system
Conceptual design
Embodiment design
Detail design
x
x
x

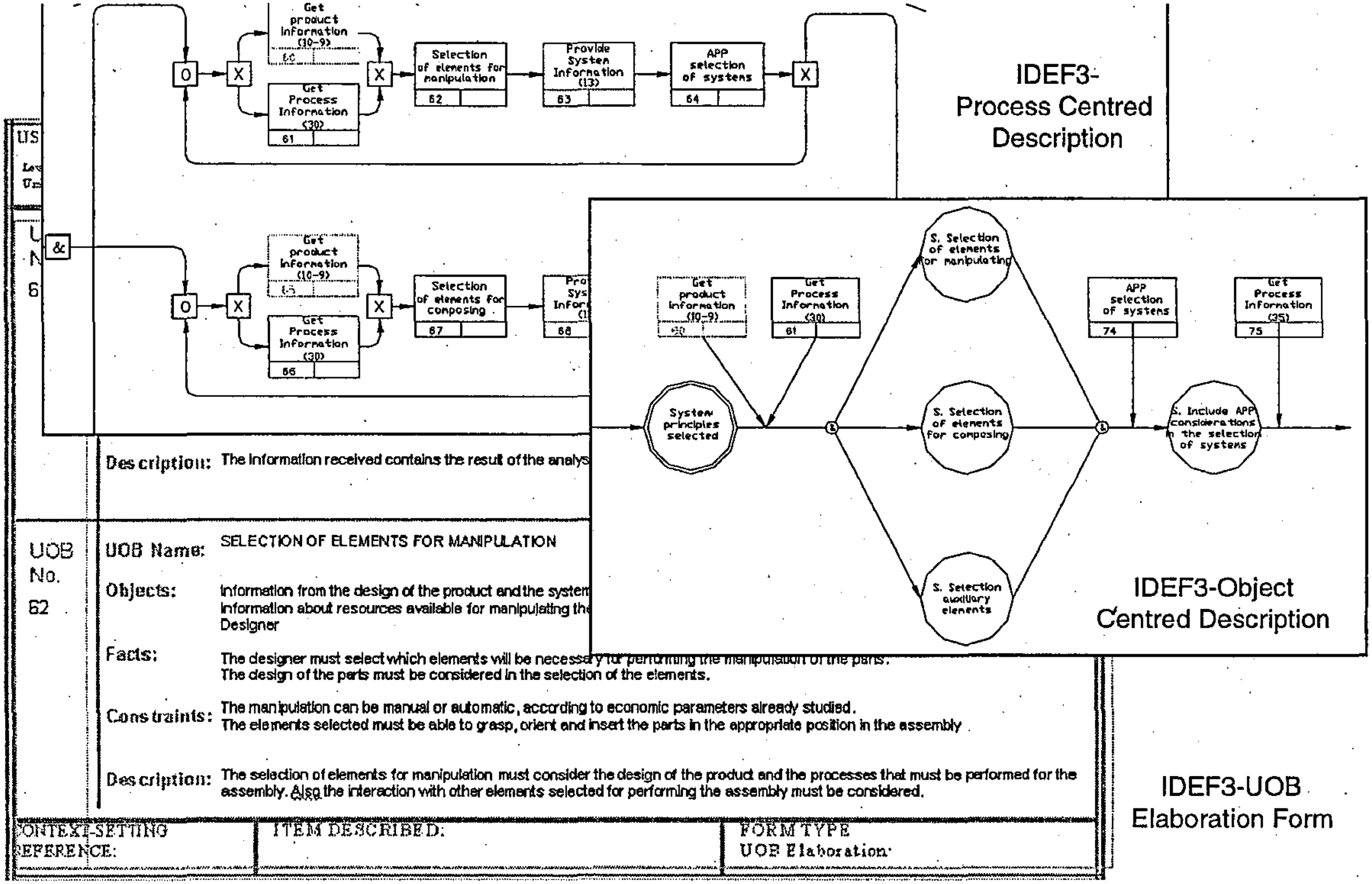



Figure 12. Relation between IDEF0 and IDEF3 methods.

.PP. A portion of the resulting diagrams is shown in gure 9.

The relation between design product, design system nd design process was established using IDEF3 process entred description diagrams. Although these diagrams re not intended for this purpose, by putting together ıe three diagrams it was possible to clearly identify the elation between them. In order to clarify those elationships, lines were traced between the diagrams, nd the corresponding numbers for the processes were ritten between parenthesis. An example is shown in gure 10.

Once the process-centred description was completed, the information was used to create the object-entred description. This describes the changes that ccur in an object through the product life cycle. The elected objects for the object-centred description were ıe product, the process and the system. This description gives a view of the transformations that take place ırough the different stages of development.

In the IDEF3 object centred description some eferents are shown between the different stages; these eferents signify the processes during which the ndicated transition occurs, or at least a process ıvolved in the transition. Due to the focus of this esearch the selected referents for this model are those elated to assembly topics.

The IDEF3 object centred descriptions were produced to display different stages in the evolution of the esign of the product, process, and system. The iagrams illustrate that some transitions require several eferents. An example of these object-centred descriptions is shown in figure 11.

In figure 12 an example of the sequence of the ıethods used is shown. The use of IDEF0 information in DEF3 is not direct, and it implies the addition of the time arameter in the model; however, the elaboration of an DEF3 process diagram starting with the information of ıe activities captured in IDEF0 has proven to be simple.

In the IDEF3 object centred description, key objects ave to be modelled in order to identify their ıteraction with processes. The elaboration forms are good way of documenting the units of behaviour, onsidering that the facts and constraints captured will e useful for determining the attributes and behaviour f their related classes in the application of the UML ıethodology.

As mentioned before, the IDEF0 and IDEF3 ıethodologies have been used to obtain the information required in the enterprise and information levels f the RM-ODP. The computational level comprises the reation of the object-oriented description of the system nd its implementation in the construction of the ıformation models and software applications.

The UML methodology was applied using a bottom-up approach, starting with the description of the general structure of the information models and the requirements that the application software should fulfil. Once the general structure of the information models was created, their decomposition in classes and subclasses was based on the information contained in the IDEF3 diagrams and unit of behaviour charts.

## 9. Use of IDEF3 UOB in the UML detailed description of the manufacturing model

As explained before, the first stage of the rational unified method is the requirements capture, this is mainly done with the application of use case diagrams that capture the needs of the users of the system (actors). However, this is done assuming that the designer has a previous good understanding of the requirements for the system. Using the information modelled with IDEF0 and IDEF3 this understanding can be provided.

In the construction of the UML class diagram it is necessary to include the classes with their attributes and behaviour, and the relationships between classes. This information can be obtained from the unit of behaviour elaboration diagrams. At this stage, the top-down and bottom-up approaches meet. An example is shown in figure 13.

## 10. Concluding comments

The approach followed is a combination of top-down and bottom-up approaches. The top-down was performed starting with the analysis of the enterprises and the definition of the basic functions that the information models have to support. This was done by modelling the activities with IDEF0 and the processes with IDEF3. These methods proved to be very effective in modelling the information at the enterprise and information levels of the RM-ODP. The combined use of these methods overcomes their individual limitations.

The bottom-up approach gave the necessary details to the general structure, detailing classes, defining their attributes and operations, and building the relationships with other classes. The UML methodology can be used alone in software design; however, it has been found that the UML methodology needs extra support in the requirements capture stage, mainly for the definition of the structure of the information models. Although there is no direct interaction between the information modelled in IDEF3 and the UML metho-

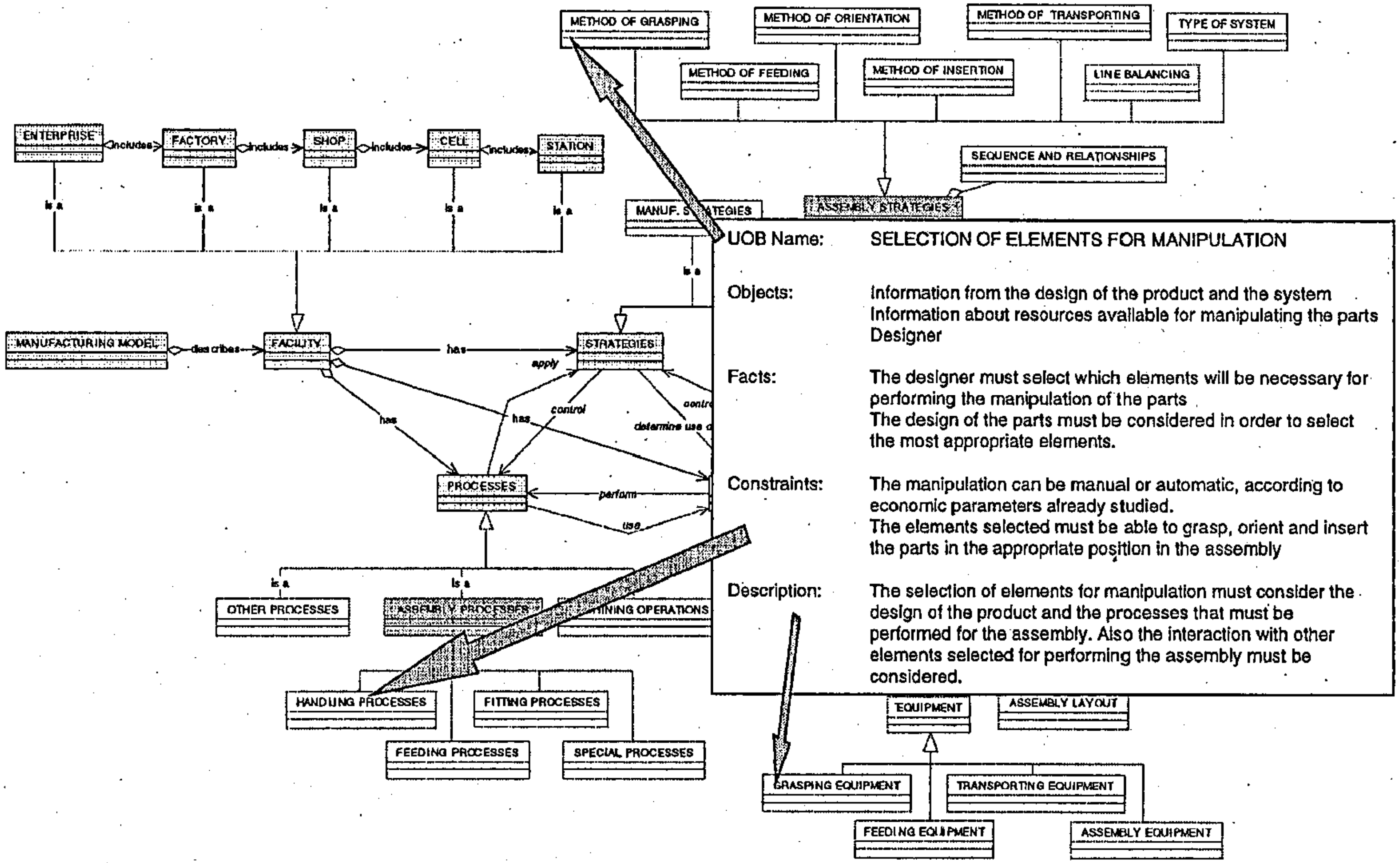



Figure 13. Use of IDEF3 UOB elaboration diagrams in the definition of subclasses in the UML class diagram of the manufacturing model.

ology, the IDEF3 modelled information can be used as guide in the definition of the attributes and behaviour f the classes.

UML notation and the rational unified process are ery useful for the definition of the application software ıat will use the information stored in the information ıodels, due to their focus on the interactions with the ser in the use cases. This incremental process allows ıe designer to build the classes and their relationships, ⁄hich are the core of the object-oriented design of the ystem.

Modelling the design process with IDEF0 and IDEF3 ives a clear view of the interaction points between DFA nd APP, besides the clear representation of the oncurrency in the design of the product, the process nd the system (resources). The definition of the DFA nd APP application software is currently under evelopment using UML, these applications will interct with a product and manufacturing model, giving upport to the designers and manufacturing engineers ırough the product life cycle.

## ıcknowledgments

The research undertaken by J.M. Dorador is funded by ıe Universidad Nacional Autónoma de México, UNAM nd by the CONACYT, México. This research is being one as part of the Manufacturing Information Models roject, funded by the EPSRC Grant No. GR/L41493.

## ‹eferences

ıSHWORTH, M., BLOOR, M. S., MCKAY, A. and OWEN, J., 1996, Adopting step for in-service configuration control. *Computers in Industry* 31, 235–253.

ıAXTER, J. E. 1994, A functional data model for assemblies. PhD thesis, Leeds University, UK.

ːOLQUHOUN, G. J. and BAINES, R. W., 1991, A generic IDEF0 model of process planning. *International Journal Production Research* 29, 2239–2257.

ɔORADOR, J. M. and YOUNG, R. I. M., 1999, Information models to support the interactions between design for assembly and assemly process planning. *International Symposium on Assembly and Task Planning*, Porto, Portugal, pp. 39–44.

DOUMEINGTS, G., 1998, GIM, *Grai Integrated Methodology.* (Dordrecht, The Netherlands: Kluwer Academic Publishers), pp. 227–288.

HUANG, C. C. and KUSIAK, A., 1998, Manufacturing control with a push-pull approach. *International Journal of Production Research* 36, 251–275.

JORYSZ, H. R. and Vernadat, F. B., 1990, CIM-OSA Part 1: Total enterprise modelling and function view. *International Journal of Computer Integrated Manufacturing* 3, 144–180.

KRAUSE, F. L., KIMURA, F., KJELLBERG, T. and LU, S. C. Y., 1993, Product modelling. *Annals of the CIRP*, 42, 695–706.

KUSIAK, A. and ZAKARIAN, A., 1996, Reliability and risk analysis of process models. *American Society of Mechanical Engineers, Materials Division* (MDv 69-2, 1995), pp 959–980.

MAYER, R. J., MENZEL, C. P., PAINTER, M. K., DEWITTE, BLINN, T., and PERAKATH, B., 1995, Information integration for concurrent engineering (IICE), IDEF3 Process Description Capture Method Report, Knowledge Based Systems Inc., USA.

MCKAY, A., BLOOR, M. S. and PENNINGTON, A., 1996, A framework for product data. *IEEE Transactions on Knowledge and Data Engineering*, 8, 825–838.

MOLINA, A., 1995, A manufacturing model to support data-driven applications for design and manufacture, PhD thesis, Loughborough University of Technology, UK.

MOSES, R. g., 1992, MOSES: Manufacturing Model & Product Model, http://leva.leeds.ac.uk/www_moses/mm.html

NIST, 1993, Integration Definition for Function Modeling (IDEF0), http://www.idef.com/downloads/Downloads.htm

NIST, 1997, The STEP Project, http://www.nist.gov/sc4/

PRASAD, B., 1996, Toward a functional design of a concurrent information modeling system. CMSE, *Computer Modeling and Simulation in Engineering* 1, 7–29.

QUATRANY, T. 1998, *Visual Modeling with Rational Rose and UML* (Reading, MA. Addison-Wesley).

TEXEL, P. P. and WILLIAMS, C. B. 1997, *Use Cases Combined with Booch/OMT/UML: Process and Products* (Engelwood Cliffs: Prentice Hall).

TOH, K. T. K., 1999, Modelling architectures: a review of their application in structured methods for information systems application. *International Journal of Production Research*, 37, 1439–1458.

WILLIAMS, T. J., 1998. *The Purdue Enterprise Reference Architecture and Methodology* (PERA). (Dordrecht, The Netherlands: Kluwer Academic Publishers), pp. 289–330.

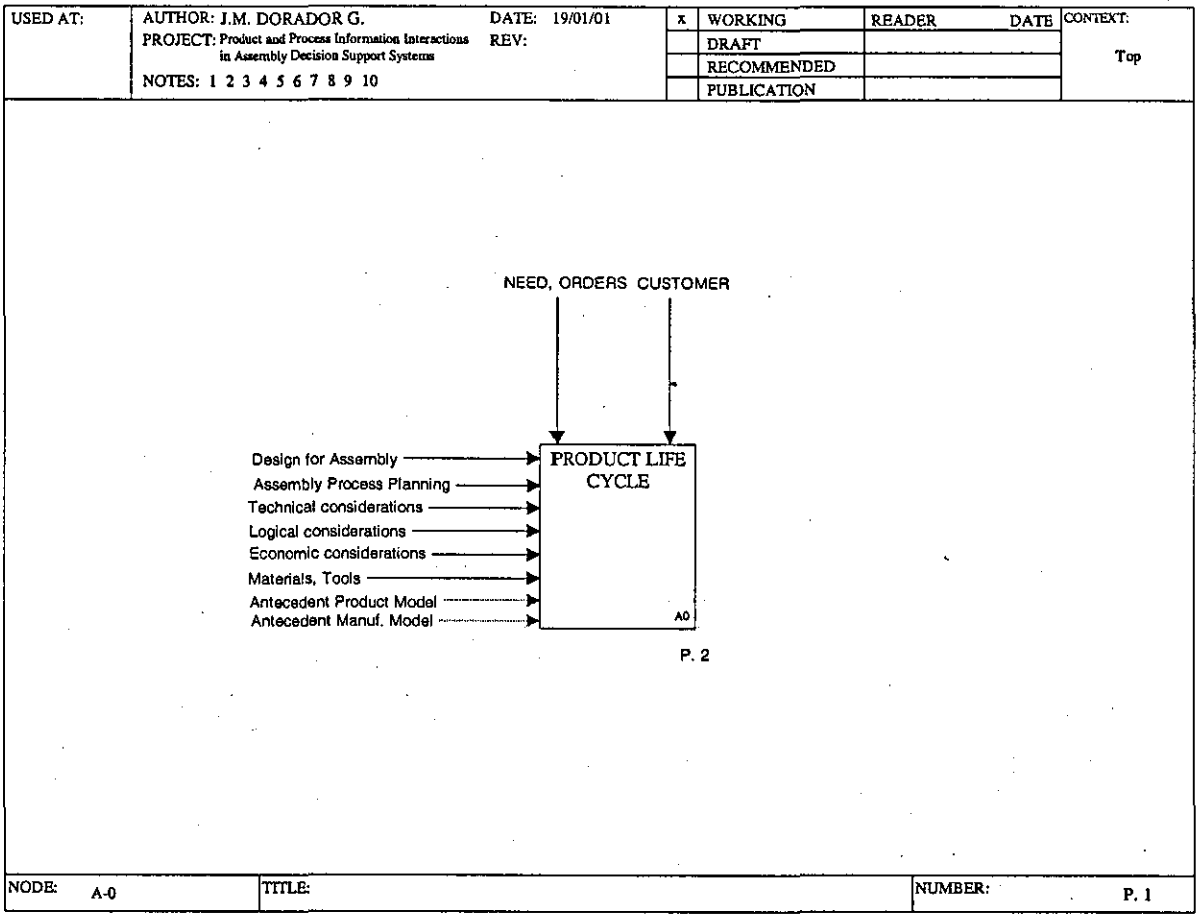
YOUNG, R. I. M. and BELL, R., 1995, Product and manufacturing models: information tools for the designer. Department of Manfuacturing Engineering, Loughborough, UK.

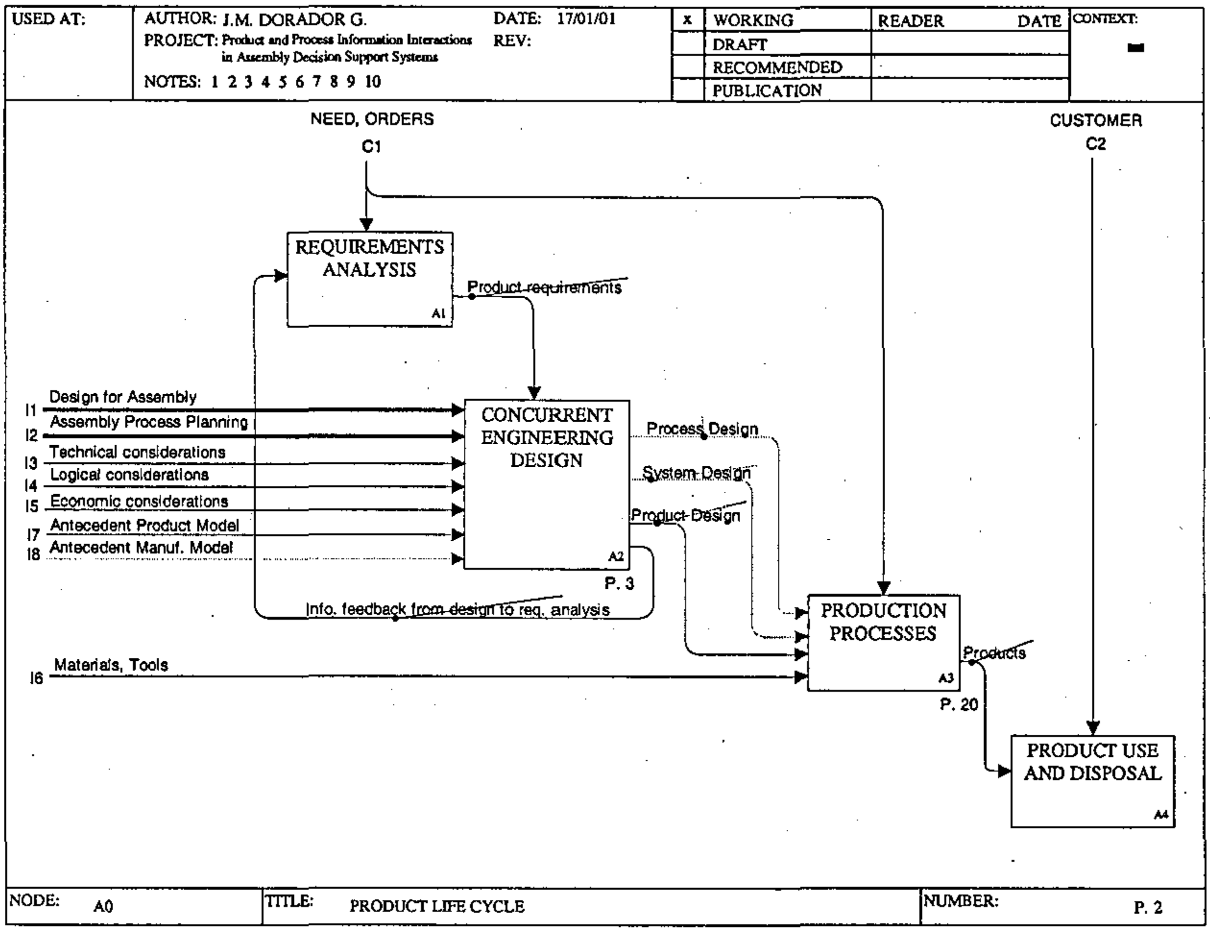
ZUOBAO, W., GUANGLENG, X. and YI, P., 1996, Product development process simulation for concurrent engineering. *High Technology Letters*, 2, 1–4.

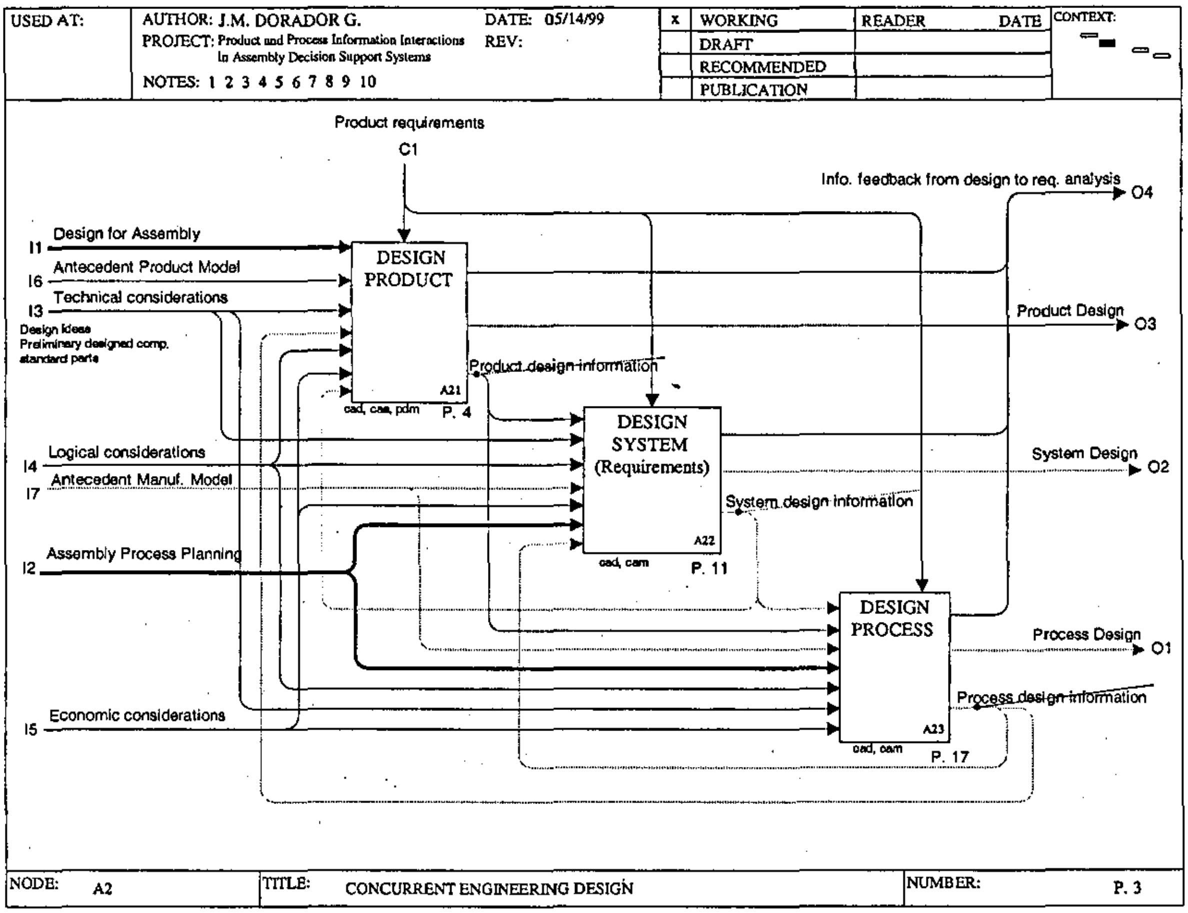
# Appendix C. IDEF0 model of the concurrent design process

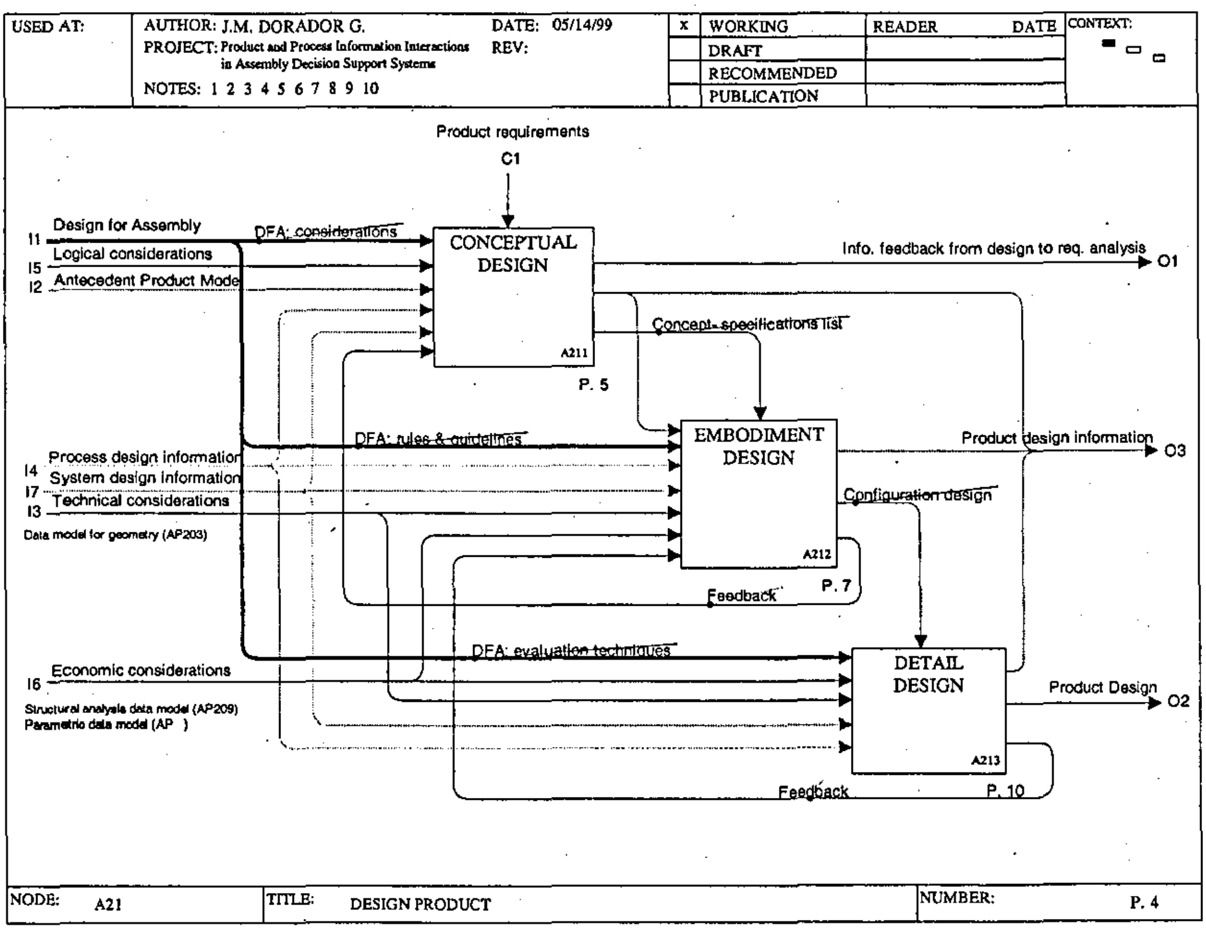
An IDEF0 model of the concurrent functions and activities performed through the product development cycle has been proposed by the author, emphasising on assembly related activities.

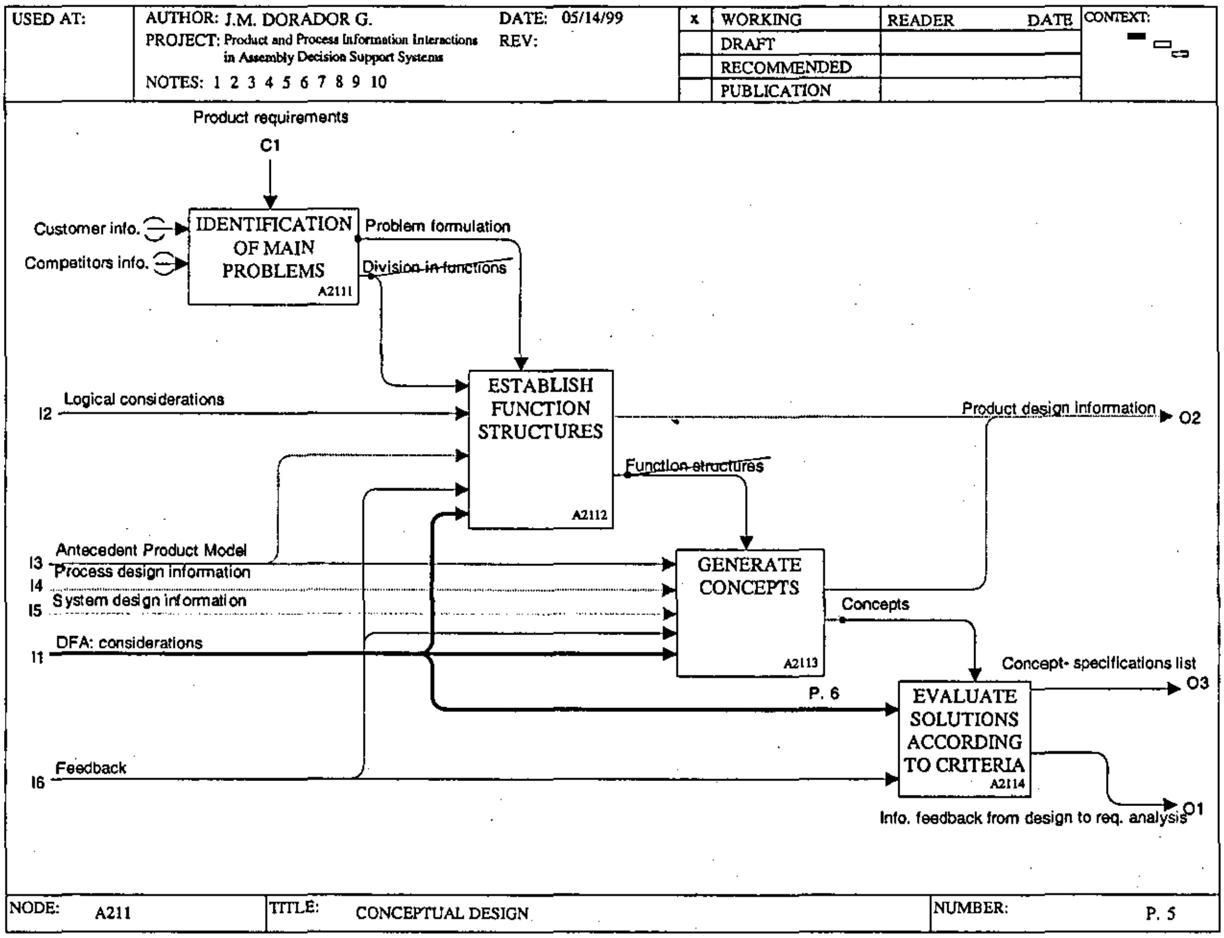
This model was divided in the following functions: Requirements Analysis, Concurrent Engineering Design, Production Processes, and Product Use and Disposal.

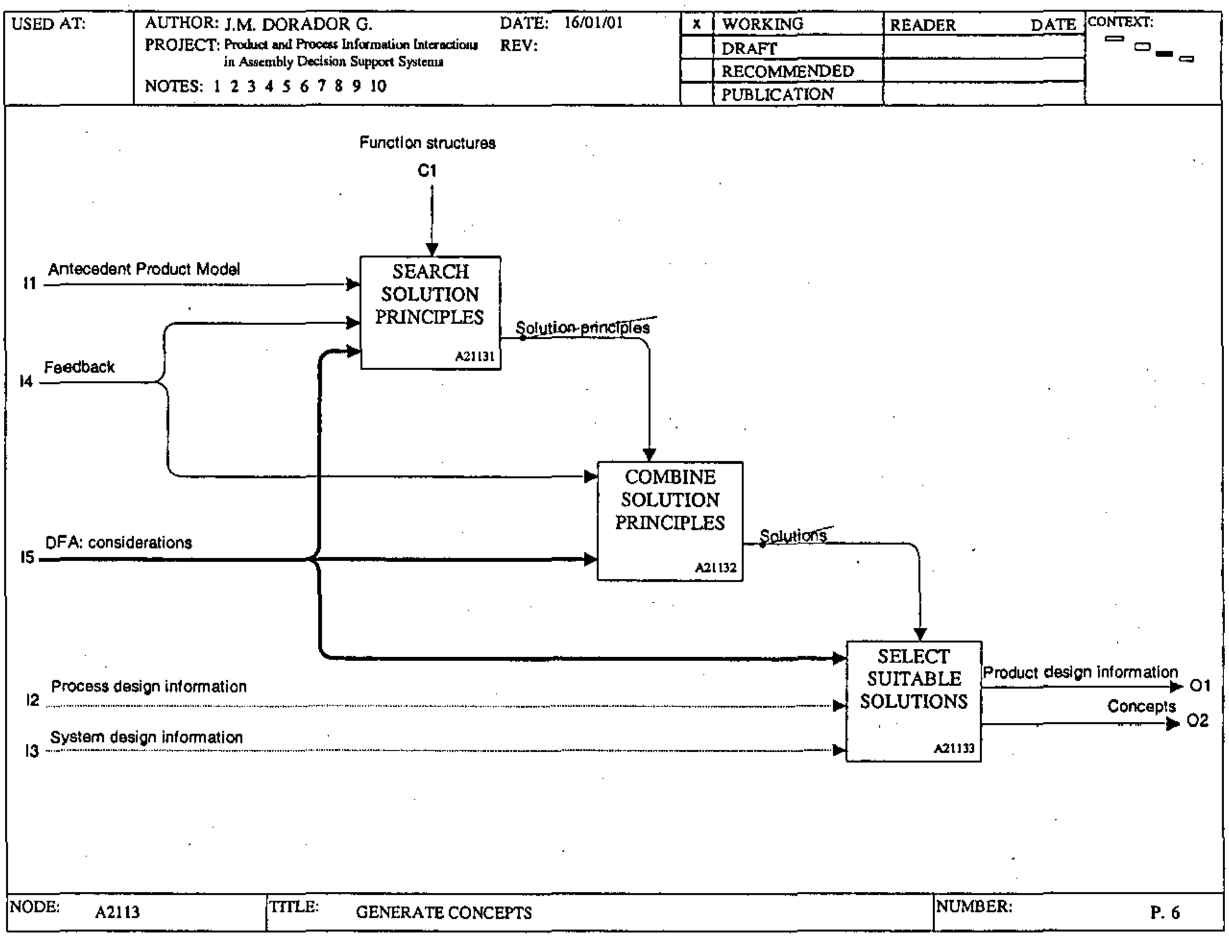

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 19/01/01
REV:
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Top
NEED, ORDERS
CUSTOMER
Design for Assembly
Assembly Process Planning
Technical considerations
Logical considerations
Economic considerations
Materials, Tools
Antecedent Product Model
Antecedent Manuf. Model
PRODUCT LIFE CYCLE
A0
P. 2
NODE: A-0
TITLE:
NUMBER: P. 1

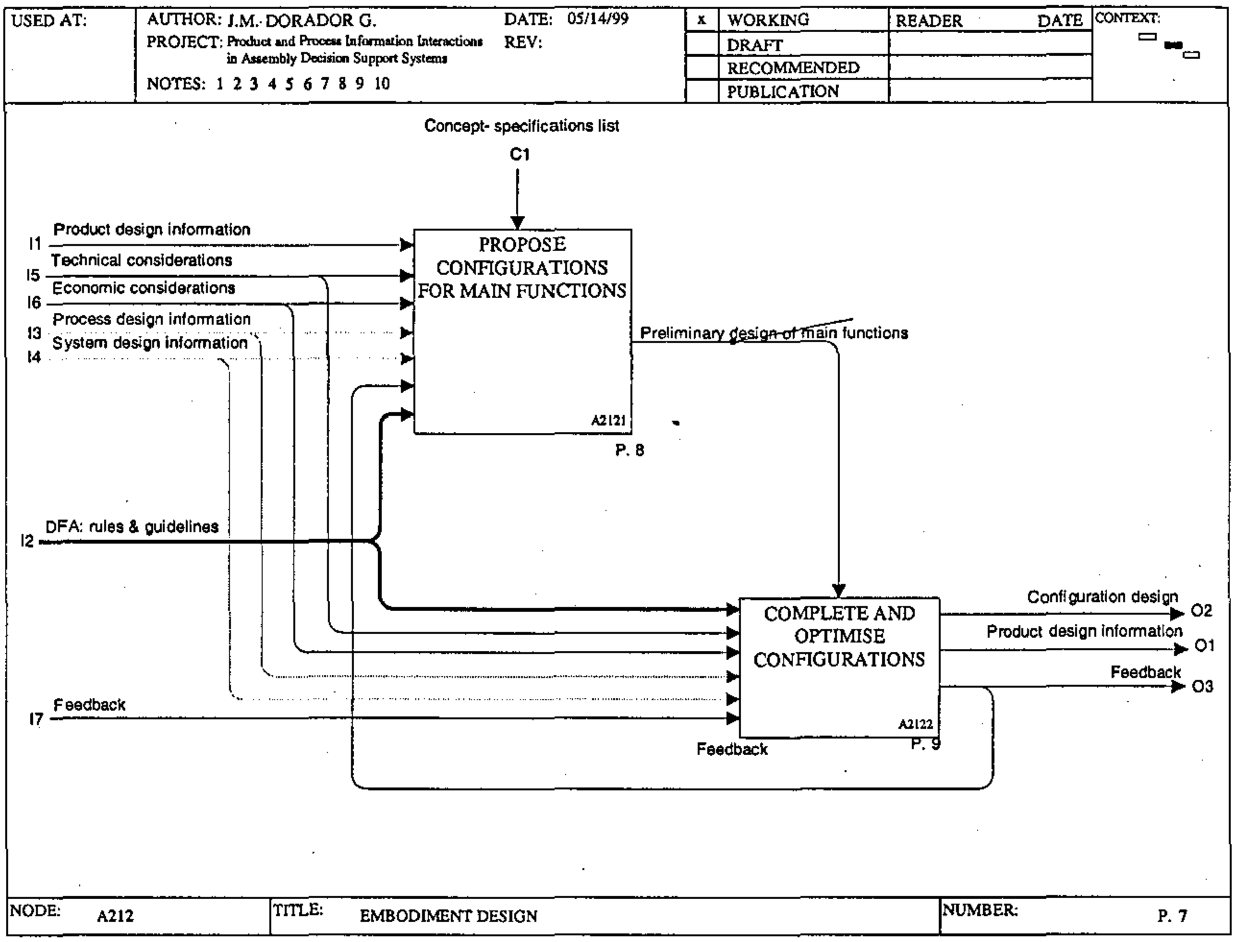

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 17/01/01
REV:
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
NEED, ORDERS
C1
CUSTOMER
C2
REQUIREMENTS ANALYSIS
A1
Product requirements
I1 Design for Assembly
I2 Assembly Process Planning
I3 Technical considerations
I4 Logical considerations
I5 Economic considerations
I7 Antecedent Product Model
I8 Antecedent Manuf. Model
CONCURRENT ENGINEERING DESIGN
A2
P. 3
Process Design
System Design
Product Design
Info. feedback from design to req. analysis
I6 Materials, Tools
PRODUCTION PROCESSES
A3
P. 20
Products
PRODUCT USE AND DISPOSAL
A4
NODE: A0
TITLE: PRODUCT LIFE CYCLE
NUMBER: P. 2

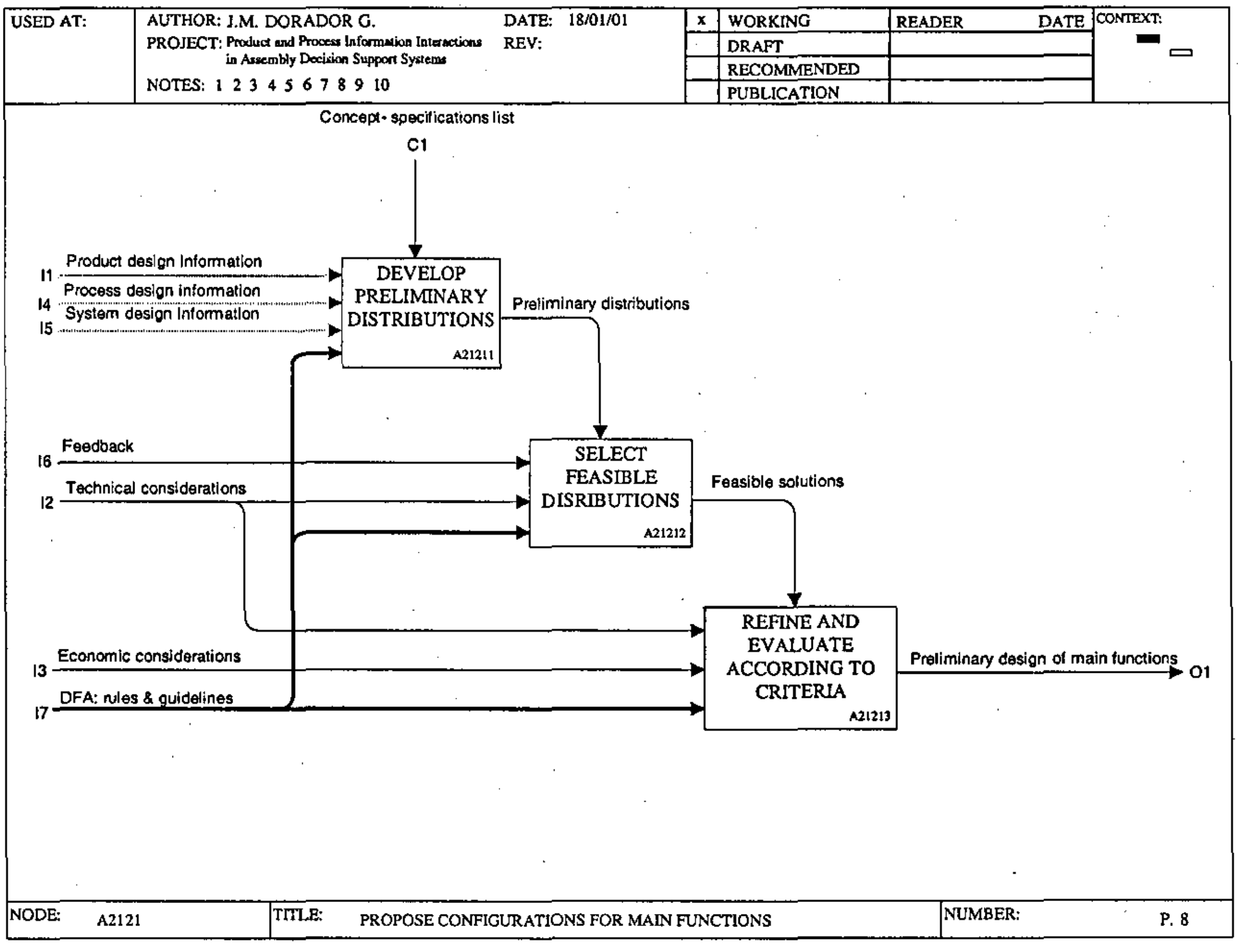

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/14/99
PROJECT: Product and Process Information Interactions In Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Info. feedback from design to req. analysis
O4
Design for Assembly
I1
Antecedent Product Model
I6
Technical considerations
I3
Design Ideas
Preliminary designed comp.
standard parts
DESIGN PRODUCT
A21
cad, cae, pdm
P. 4
Product Design
O3
Product design information
DESIGN SYSTEM (Requirements)
A22
cad, cam
P. 11
Logical considerations
I4
Antecedent Manuf. Model
I7
System Design
O2
System design information
Assembly Process Planning
I2
DESIGN PROCESS
A23
cad, cam
P. 17
Process Design
O1
Process design information
Economic considerations
I5
NODE: A2
TITLE: CONCURRENT ENGINEERING DESIGN
NUMBER: P. 3

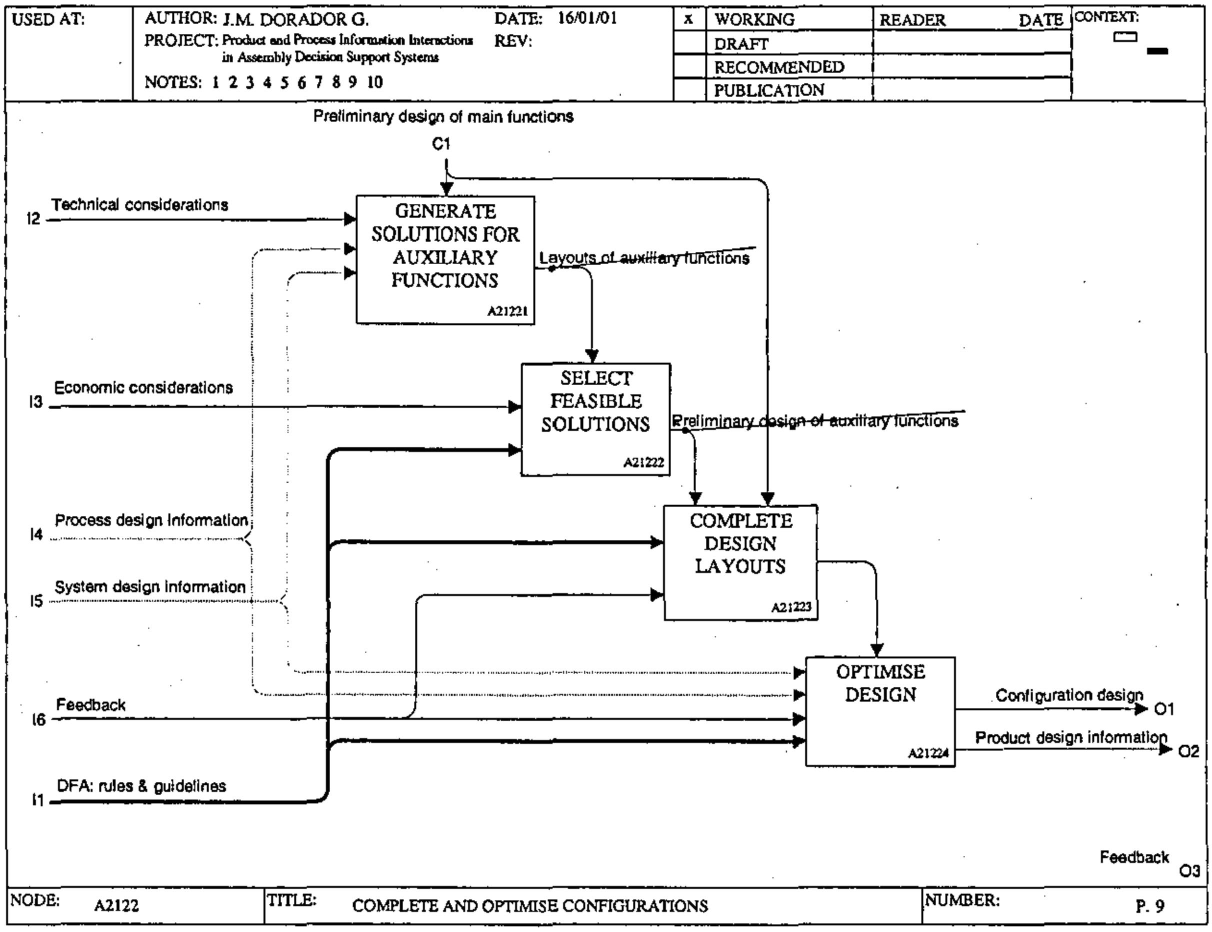

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/14/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Design for Assembly
I1
Logical considerations
I5
Antecedent Product Mode
I2
DFA: considerations
CONCEPTUAL DESIGN
A211
P. 5
Info. feedback from design to req. analysis
O1
Concept- specifications list
DFA: rules & guidelines
EMBODIMENT DESIGN
A212
P. 7
Product design information
O3
Process design informatio
I4
System design informatio
I7
Technical considerations
I3
Data model for geometry (AP203)
Configuration design
Feedback
DFA: evaluation techniques
Economic considerations
I6
Structural analysis data model (AP209)
Parametric data model (AP )
DETAIL DESIGN
A213
P. 10
Product Design
O2
Feedback
NODE: A21
TITLE: DESIGN PRODUCT
NUMBER: P. 4

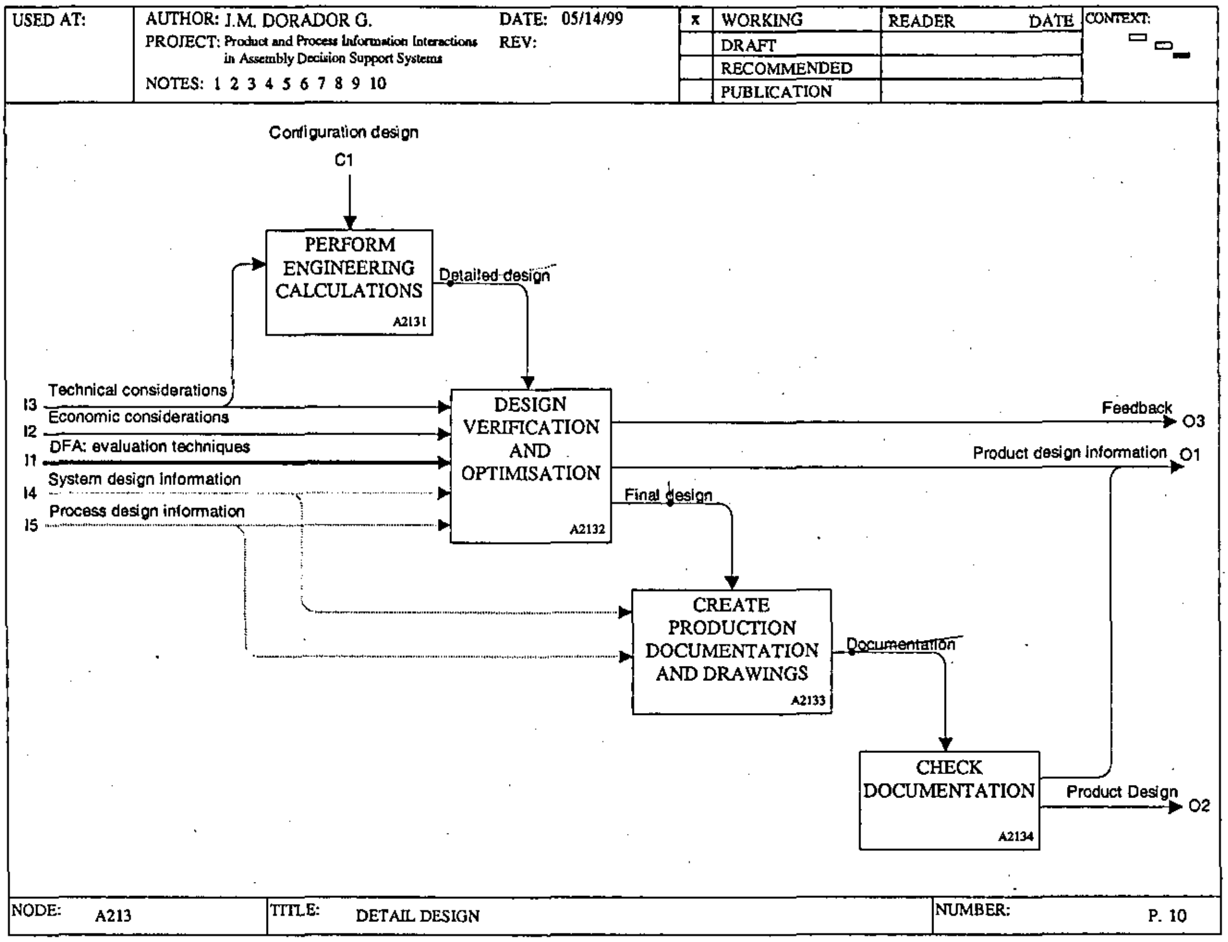

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 05/14/99
REV:
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Customer info.
Competitors info.
IDENTIFICATION OF MAIN PROBLEMS
A2111
Problem formulation
Division in functions
ESTABLISH FUNCTION STRUCTURES
A2112
Logical considerations
I2
Product design information
O2
Function structures
Antecedent Product Model
I3
Process design information
I4
System design information
I5
DFA: considerations
I1
GENERATE CONCEPTS
A2113
Concepts
P. 6
Concept- specifications list
O3
EVALUATE SOLUTIONS ACCORDING TO CRITERIA
A2114
Feedback
I6
Info. feedback from design to req. analysis
O1
NODE: A211
TITLE: CONCEPTUAL DESIGN
NUMBER: P. 5

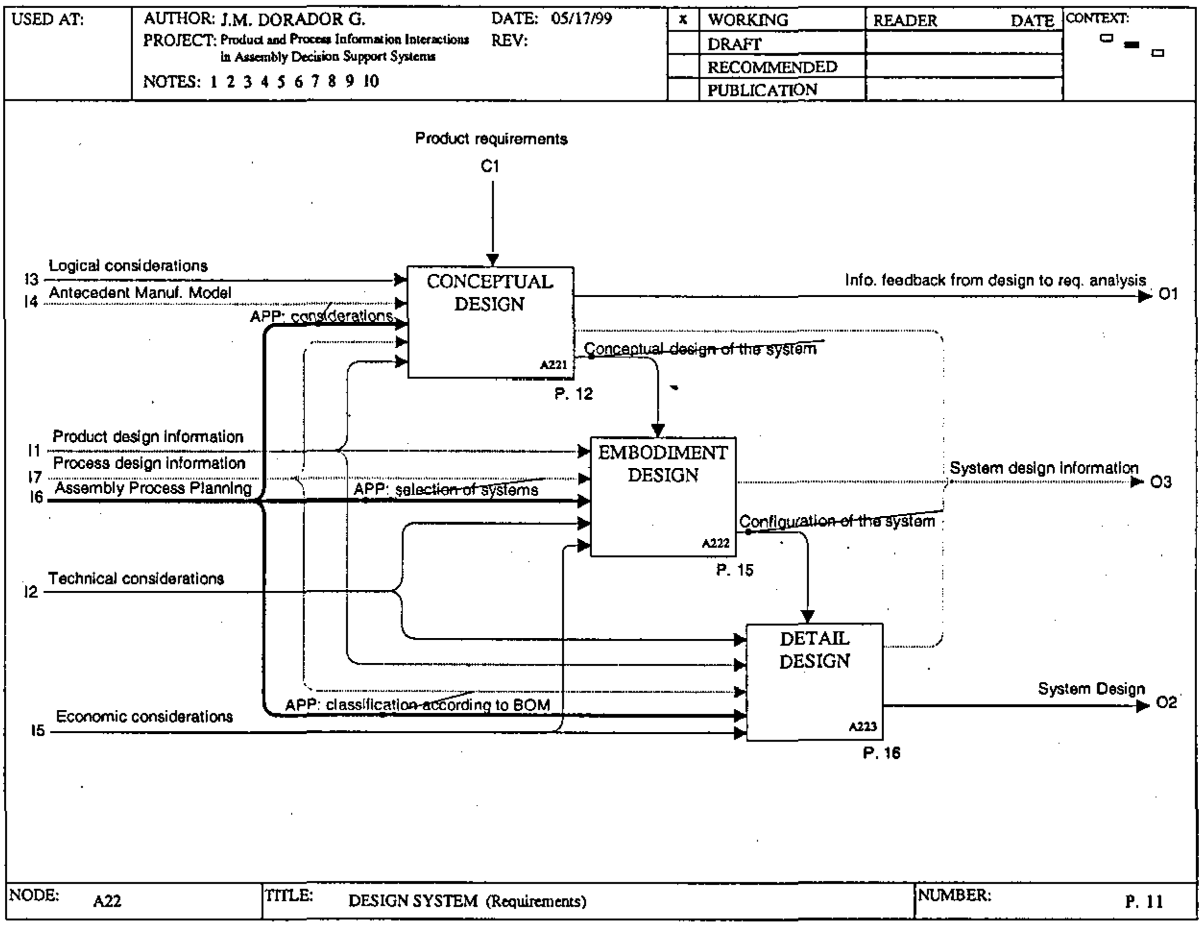

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 16/01/01
REV:
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Function structures
C1
Antecedent Product Model
I1
SEARCH SOLUTION PRINCIPLES
A21131
Solution principles
Feedback
I4
COMBINE SOLUTION PRINCIPLES
A21132
Solutions
DFA: considerations
I5
SELECT SUITABLE SOLUTIONS
A21133
Product design information
O1
Concepts
O2
Process design information
I2
System design information
I3
NODE: A2113
TITLE: GENERATE CONCEPTS
NUMBER: P. 6

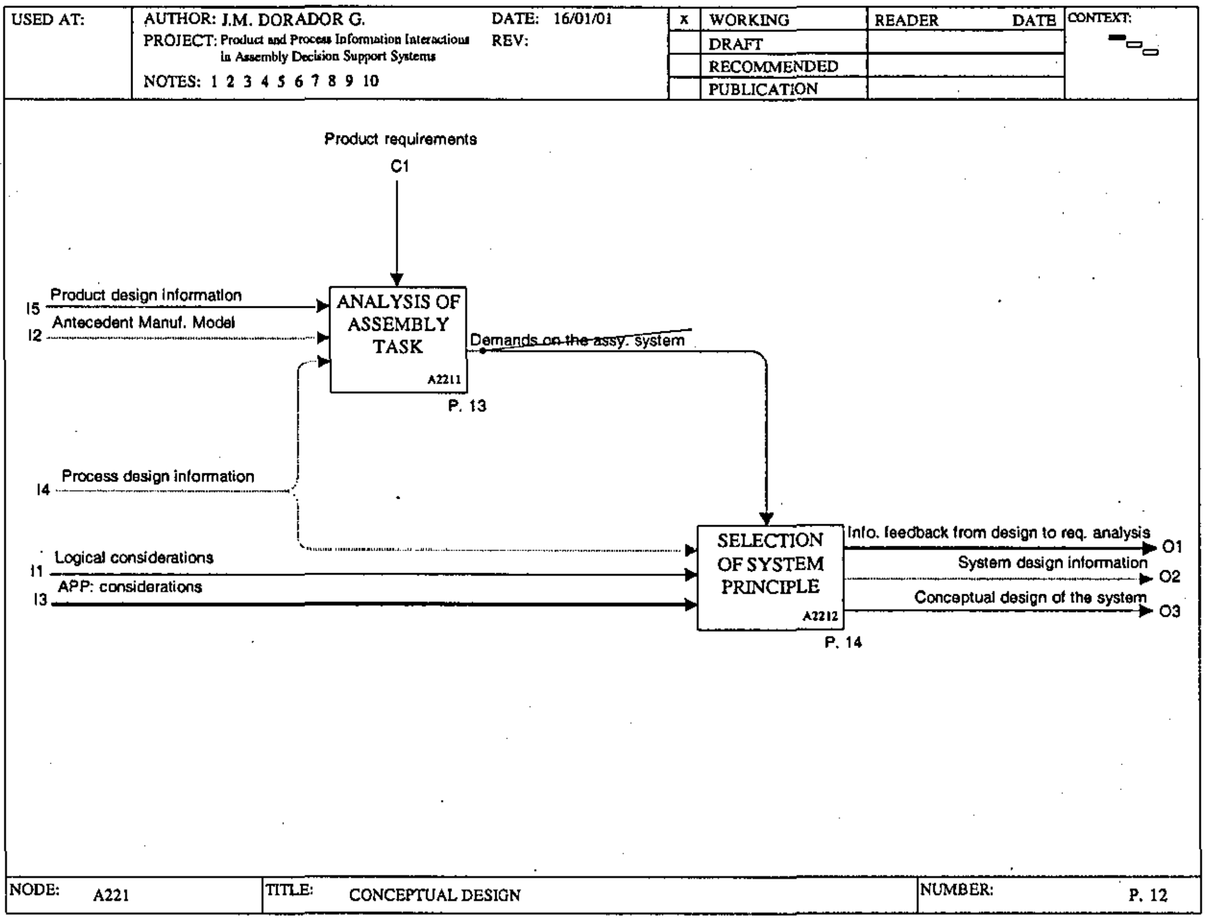

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/14/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Concept- specifications list
C1
Product design information
I1
Technical considerations
I5
Economic considerations
I6
Process design information
I3
System design information
I4
PROPOSE CONFIGURATIONS FOR MAIN FUNCTIONS
A2121
P. 8
Preliminary design of main functions
DFA: rules & guidelines
I2
Feedback
I7
COMPLETE AND OPTIMISE CONFIGURATIONS
A2122
P. 9
Feedback
Configuration design
O2
Product design information
O1
Feedback
O3
NODE: A212
TITLE: EMBODIMENT DESIGN
NUMBER: P. 7

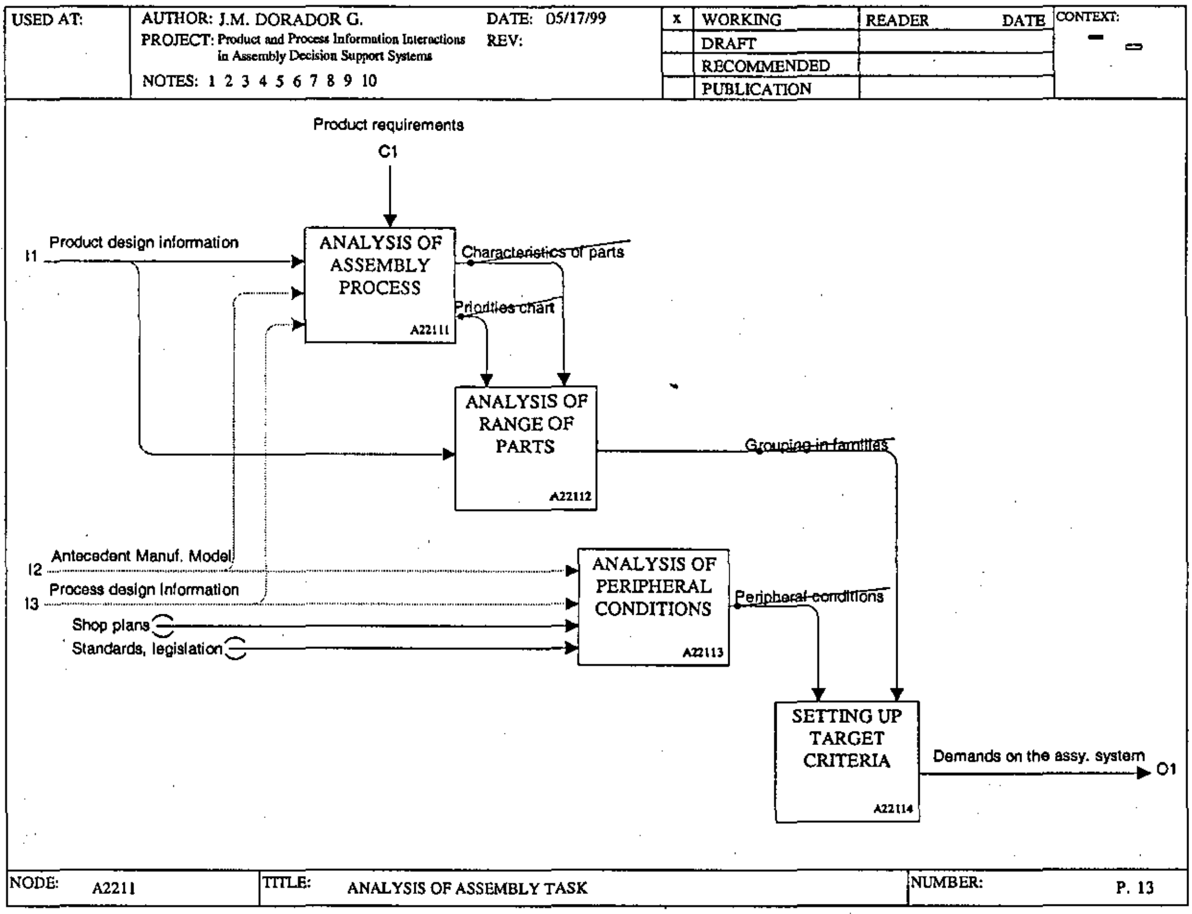

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 18/01/01
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Concept- specifications list
C1
Product design information
I1
Process design information
I4
System design information
I5
DEVELOP PRELIMINARY DISTRIBUTIONS
A21211
Preliminary distributions
Feedback
I6
Technical considerations
I2
SELECT FEASIBLE DISRIBUTIONS
A21212
Feasible solutions
Economic considerations
I3
DFA: rules & guidelines
I7
REFINE AND EVALUATE ACCORDING TO CRITERIA
A21213
Preliminary design of main functions
O1
NODE: A2121
TITLE: PROPOSE CONFIGURATIONS FOR MAIN FUNCTIONS
NUMBER: P. 8

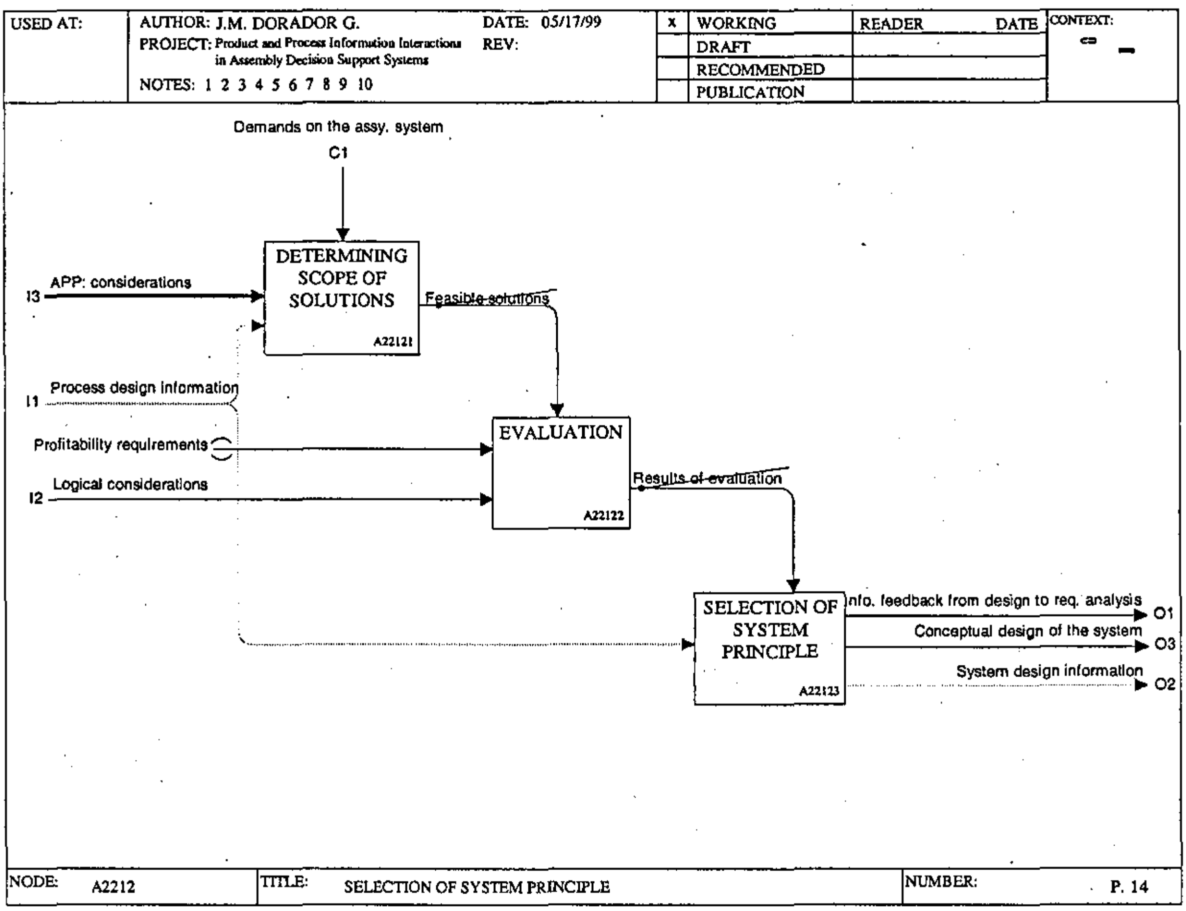

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 16/01/01
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Preliminary design of main functions
C1
Technical considerations
I2
GENERATE SOLUTIONS FOR AUXILIARY FUNCTIONS
A21221
Layouts of auxiliary functions
Economic considerations
I3
SELECT FEASIBLE SOLUTIONS
A21222
Preliminary design of auxiliary functions
Process design information
I4
System design information
I5
COMPLETE DESIGN LAYOUTS
A21223
OPTIMISE DESIGN
A21224
Feedback
I6
Configuration design
O1
Product design information
O2
DFA: rules & guidelines
I1
Feedback
O3
NODE: A2122
TITLE: COMPLETE AND OPTIMISE CONFIGURATIONS
NUMBER: P. 9

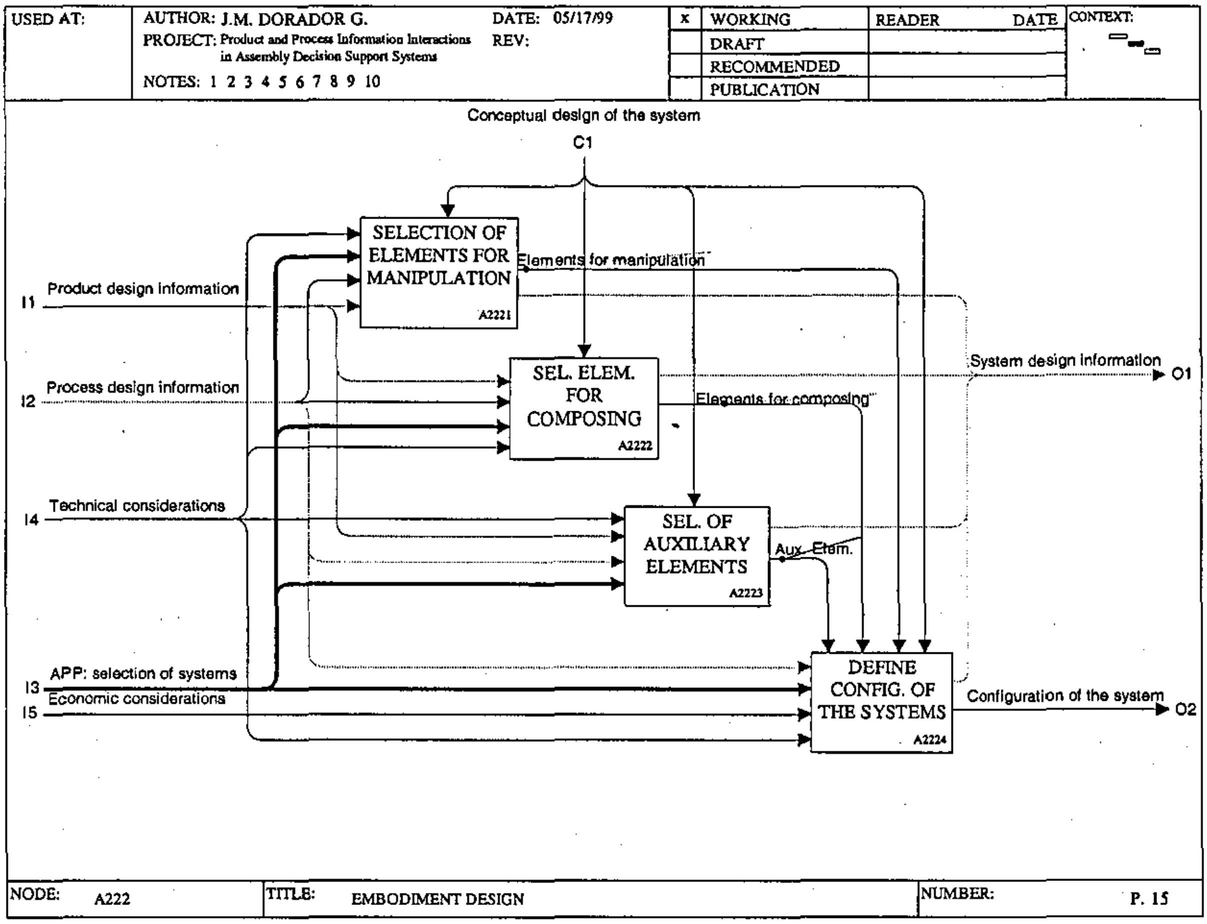

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/14/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Configuration design
C1
PERFORM ENGINEERING CALCULATIONS
A2131
Detailed design
Technical considerations
I3
Economic considerations
I2
DFA: evaluation techniques
I1
System design information
I4
Process design information
I5
DESIGN VERIFICATION AND OPTIMISATION
A2132
Feedback
O3
Product design information
O1
Final design
CREATE PRODUCTION DOCUMENTATION AND DRAWINGS
A2133
Documentation
CHECK DOCUMENTATION
A2134
Product Design
O2
NODE: A213
TITLE: DETAIL DESIGN
NUMBER: P. 10

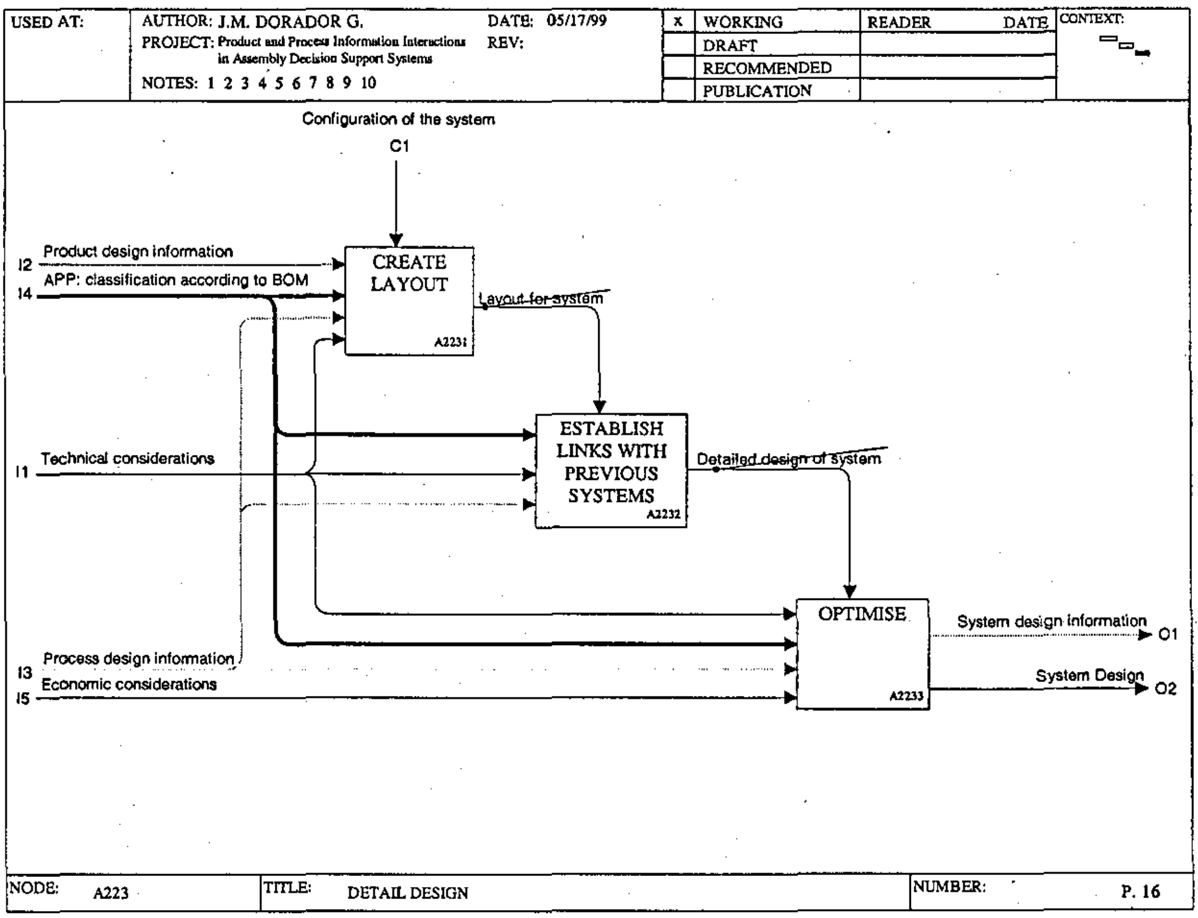

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/17/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Logical considerations
I3
Antecedent Manuf. Model
I4
APP: considerations
CONCEPTUAL DESIGN
A221
P. 12
Info. feedback from design to req. analysis
O1
Conceptual design of the system
Product design information
I1
Process design information
I7
Assembly Process Planning
I6
APP: selection of systems
EMBODIMENT DESIGN
A222
P. 15
System design information
O3
Configuration of the system
Technical considerations
I2
DETAIL DESIGN
A223
P. 16
System Design
O2
APP: classification according to BOM
Economic considerations
I5
NODE: A22
TITLE: DESIGN SYSTEM (Requirements)
NUMBER: P. 11

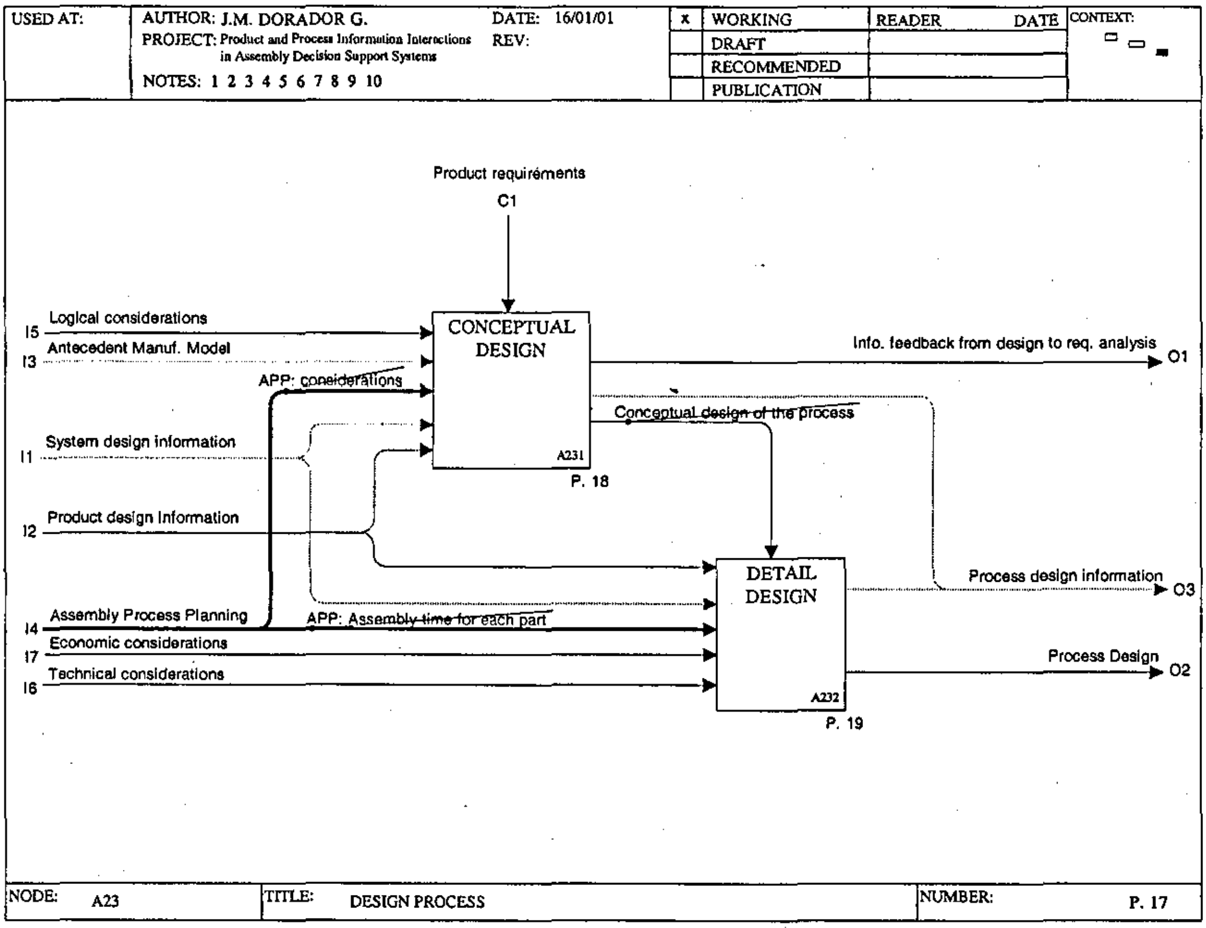

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 16/01/01
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Product design information
I5
Antecedent Manuf. Model
I2
ANALYSIS OF ASSEMBLY TASK
A2211
P. 13
Demands on the assy. system
Process design information
I4
Logical considerations
I1
APP: considerations
I3
SELECTION OF SYSTEM PRINCIPLE
A2212
P. 14
Info. feedback from design to req. analysis
O1
System design information
O2
Conceptual design of the system
O3
NODE: A221
TITLE: CONCEPTUAL DESIGN
NUMBER: P. 12

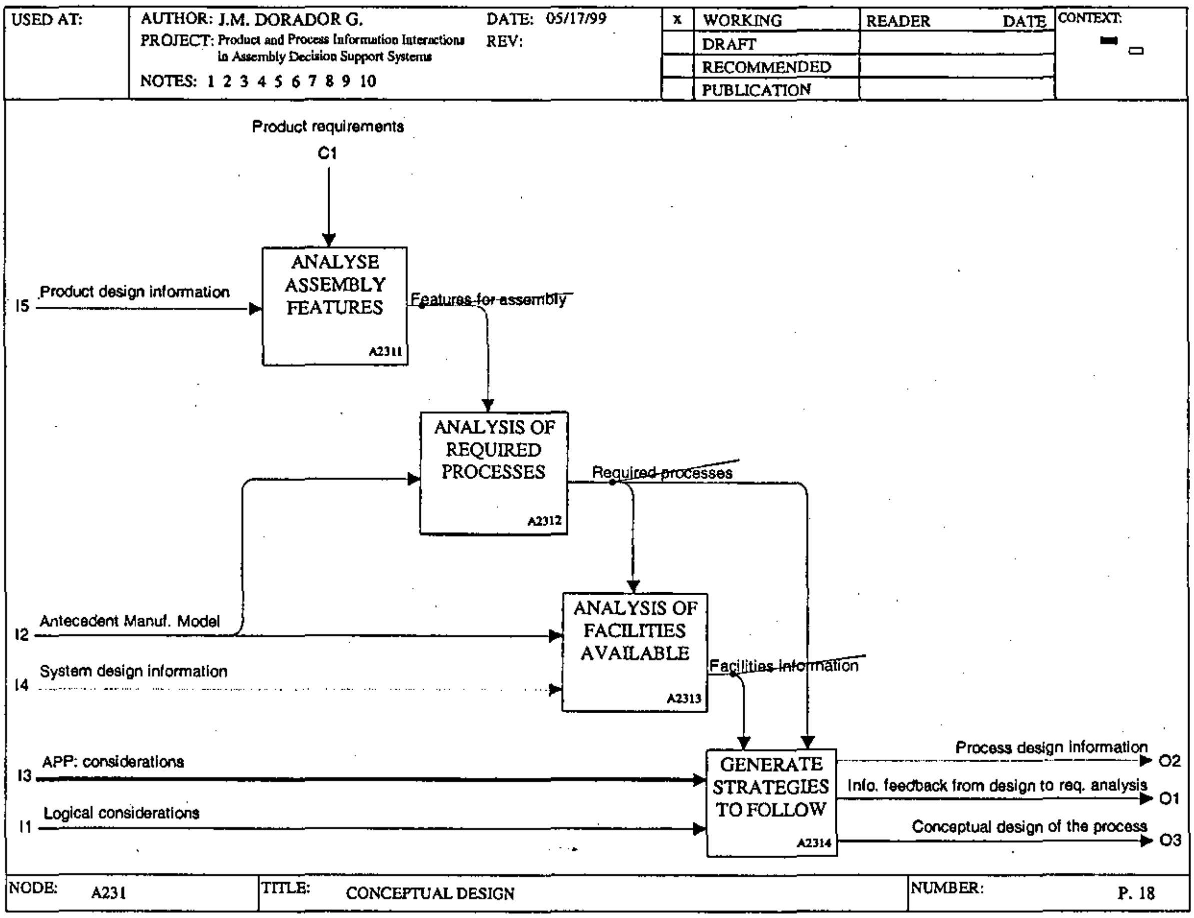

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 05/17/99
REV:
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
I1
Product design information
ANALYSIS OF ASSEMBLY PROCESS
A22111
Characteristics of parts
Priorities chart
ANALYSIS OF RANGE OF PARTS
A22112
Grouping in families
I2
Antecedent Manuf. Model
I3
Process design information
Shop plans
Standards, legislation
ANALYSIS OF PERIPHERAL CONDITIONS
A22113
Peripheral conditions
SETTING UP TARGET CRITERIA
A22114
Demands on the assy. system
O1
NODE: A2211
TITLE: ANALYSIS OF ASSEMBLY TASK
NUMBER: P. 13

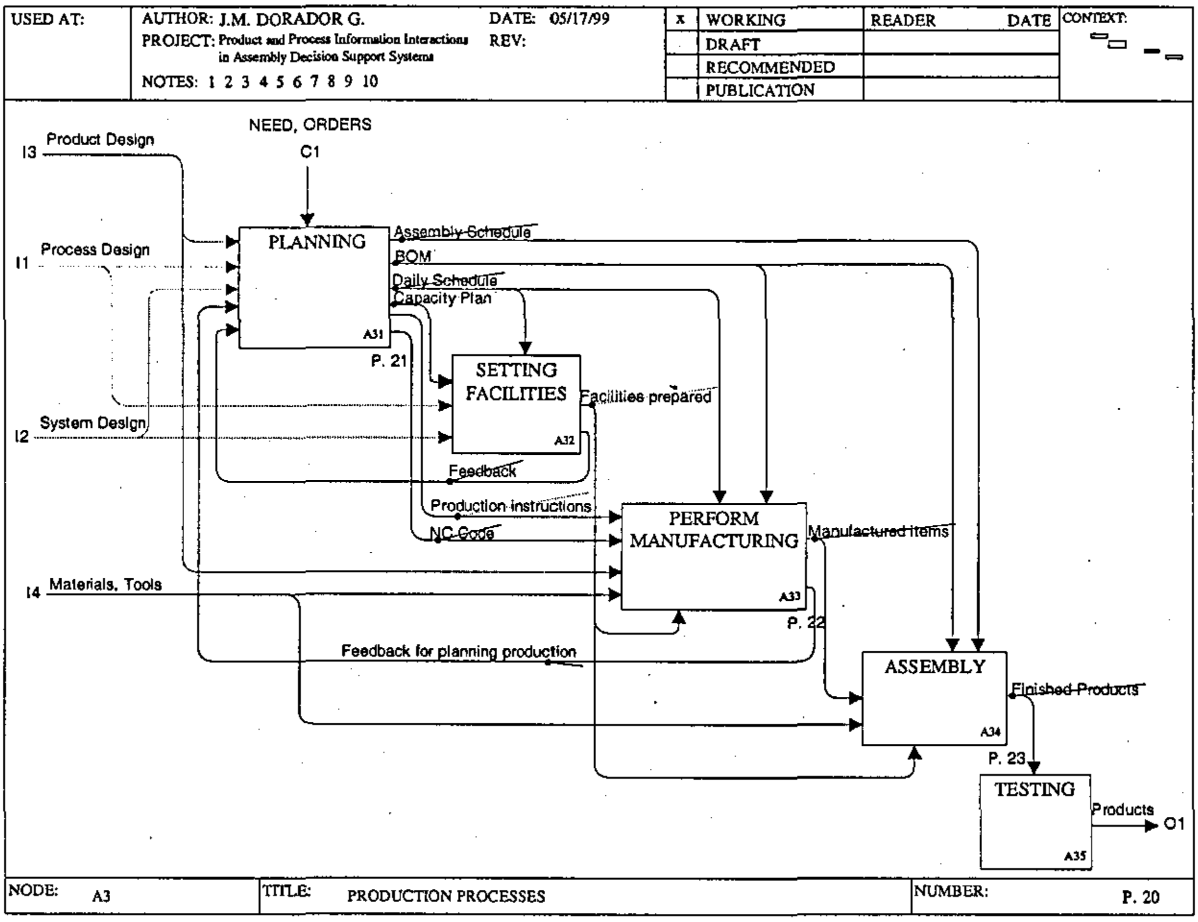

USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 05/17/99
REV:
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Demands on the assy. system
C1
I3
APP: considerations
DETERMINING SCOPE OF SOLUTIONS
A22121
Feasible solutions
I1
Process design information
Profitability requirements
I2
Logical considerations
EVALUATION
A22122
Results of evaluation
SELECTION OF SYSTEM PRINCIPLE
A22123
Info. feedback from design to req. analysis
O1
Conceptual design of the system
O3
System design information
O2
NODE: A2212
TITLE: SELECTION OF SYSTEM PRINCIPLE
NUMBER: P. 14

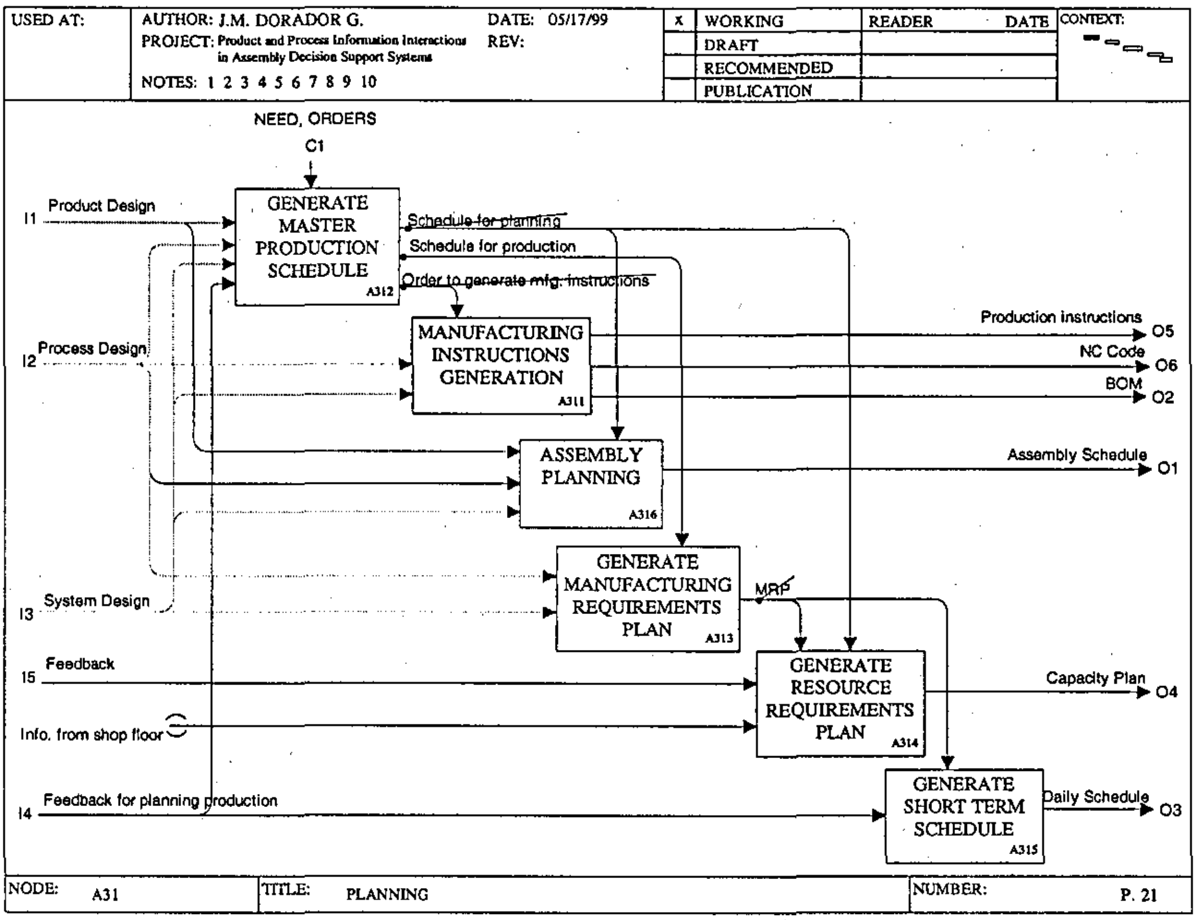

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/17/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Conceptual design of the system
C1
SELECTION OF ELEMENTS FOR MANIPULATION
A2221
Elements for manipulation
I1 Product design information
I2 Process design information
SEL. ELEM. FOR COMPOSING
A2222
Elements for composing
System design information
O1
I4 Technical considerations
SEL. OF AUXILIARY ELEMENTS
A2223
Aux. Elem.
I3 APP: selection of systems
I5 Economic considerations
DEFINE CONFIG. OF THE SYSTEMS
A2224
Configuration of the system
O2
NODE: A222
TITLE: EMBODIMENT DESIGN
NUMBER:
P. 15

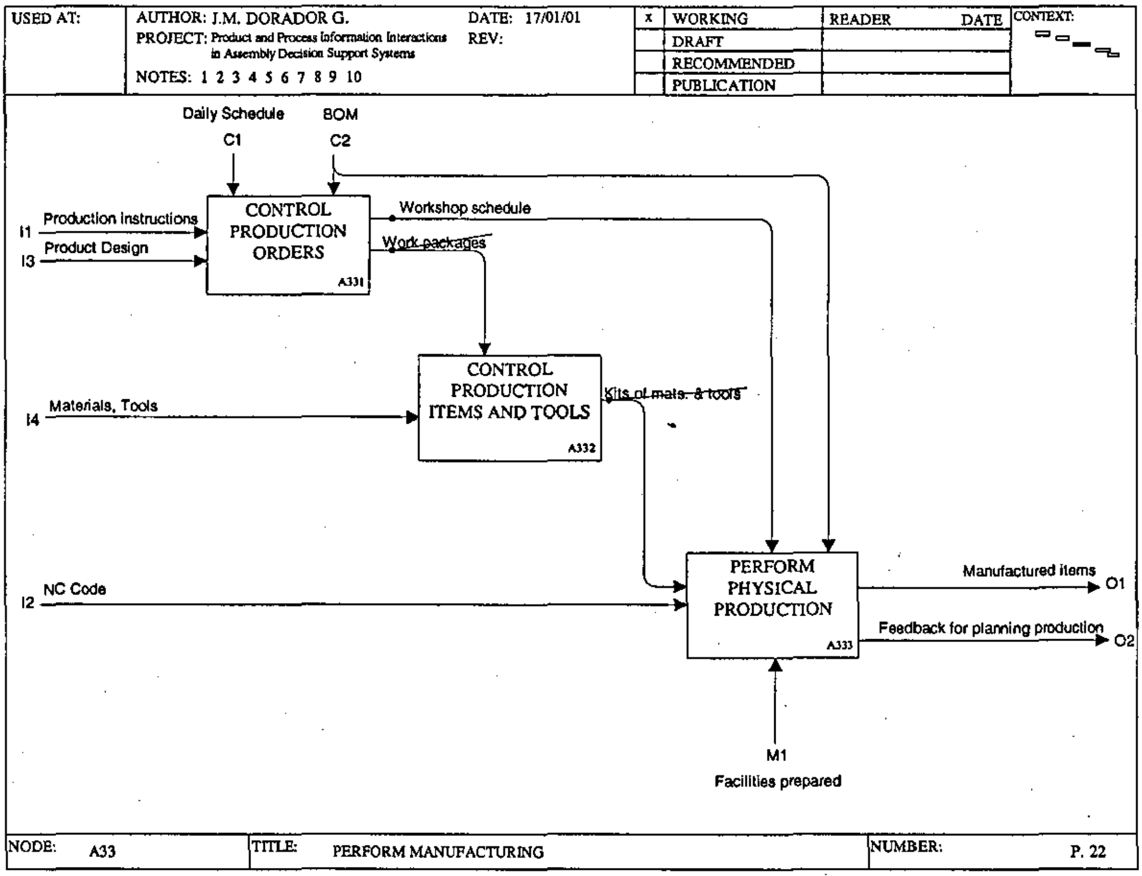

USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/17/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Configuration of the system
C1
I2 Product design information
I4 APP: classification according to BOM
CREATE LAYOUT
A2231
Layout for system
I1 Technical considerations
ESTABLISH LINKS WITH PREVIOUS SYSTEMS
A2232
Detailed design of system
OPTIMISE
A2233
System design information
O1
I3 Process design information
I5 Economic considerations
System Design
O2
NODE: A223
TITLE: DETAIL DESIGN
NUMBER:
P. 16


USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 16/01/01
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Logical considerations
I5
Antecedent Manuf. Model
I3
APP: considerations
System design information
I1
Product design Information
I2
Assembly Process Planning
I4
APP: Assembly time for each part
Economic considerations
I7
Technical considerations
I6
CONCEPTUAL DESIGN
A231
P. 18
Info. feedback from design to req. analysis
O1
Conceptual design of the process
DETAIL DESIGN
A232
P. 19
Process design information
O3
Process Design
O2
NODE: A23
TITLE: DESIGN PROCESS
NUMBER: P. 17


USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 05/17/99
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Product requirements
C1
Product design information
I5
ANALYSE ASSEMBLY FEATURES
A2311
Features for assembly
ANALYSIS OF REQUIRED PROCESSES
A2312
Required processes
Antecedent Manuf. Model
I2
System design information
I4
ANALYSIS OF FACILITIES AVAILABLE
A2313
Facilities information
APP: considerations
I3
Logical considerations
I1
GENERATE STRATEGIES TO FOLLOW
A2314
Process design information
O2
Info. feedback from design to req. analysis
O1
Conceptual design of the process
O3
NODE: A231
TITLE: CONCEPTUAL DESIGN
NUMBER: P. 18


USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
DATE: 05/17/99
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
NEED, ORDERS
C1
I3 Product Design
I1 Process Design
I2 System Design
I4 Materials, Tools
PLANNING
A31
P. 21
Assembly Schedule
BOM
Daily Schedule
Capacity Plan
SETTING FACILITIES
A32
Facilities prepared
Feedback
Production instructions
NC Code
PERFORM MANUFACTURING
A33
P. 22
Manufactured items
Feedback for planning production
ASSEMBLY
A34
P. 23
Finished Products
TESTING
A35
Products
O1
NODE: A3
TITLE: PRODUCTION PROCESSES
NUMBER: P. 20


USED AT:
AUTHOR: J.M. DORADOR G.
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
DATE: 05/17/99
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
NEED, ORDERS
C1
I1 Product Design
I2 Process Design
I3 System Design
I5 Feedback
Info. from shop floor
I4 Feedback for planning production
GENERATE MASTER PRODUCTION SCHEDULE
A312
Schedule for planning
Schedule for production
Order to generate mfg. instructions
MANUFACTURING INSTRUCTIONS GENERATION
A311
Production instructions
O5
NC Code
O6
BOM
O2
ASSEMBLY PLANNING
A316
Assembly Schedule
O1
GENERATE MANUFACTURING REQUIREMENTS PLAN
A313
MRP
GENERATE RESOURCE REQUIREMENTS PLAN
A314
Capacity Plan
O4
GENERATE SHORT TERM SCHEDULE
A315
Daily Schedule
O3
NODE: A31
TITLE: PLANNING
NUMBER: P. 21


USED AT:
AUTHOR: J.M. DORADOR G.
DATE: 17/01/01
PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
x
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Daily Schedule
BOM
C1
C2
Production Instructions
I1
Product Design
I3
CONTROL PRODUCTION ORDERS
A331
Workshop schedule
Work packages
CONTROL PRODUCTION ITEMS AND TOOLS
A332
Materials, Tools
I4
Kits of mats. & tools
PERFORM PHYSICAL PRODUCTION
A333
NC Code
I2
Manufactured items
O1
Feedback for planning production
O2
M1
Facilities prepared
NODE: A33
TITLE: PERFORM MANUFACTURING
NUMBER: P. 22

# Appendix D. IDEF3 model of the concurrent design process

An IDEF3 model of the processes performed through the concurrent design of the product, processes and resources was proposed by the author in order to find the information requirements necessary to support Design For Assembly and Assembly Process Planning. This model was done as part of the methodology followed by the author in this research, explained in chapter 3.

The first section of this appendix includes the process and object centred descriptions for the IDEF3 model of the concurrent design of the product, process and resources.

Section D.2 includes the decomposition of the Units of Behaviour that are specifically related to the Design For Assembly and Assembly Process Planning processes that influence directly the concurrent design process. This section also includes the Elaboration Diagrams for these Units of Behaviour.

## D.1. IDEF3 Model of the Concurrent Design of the Product, Process and Resources

### D.1.1. IDEF3 Process Centred Description

#### *D.1.1.1. Product Design*

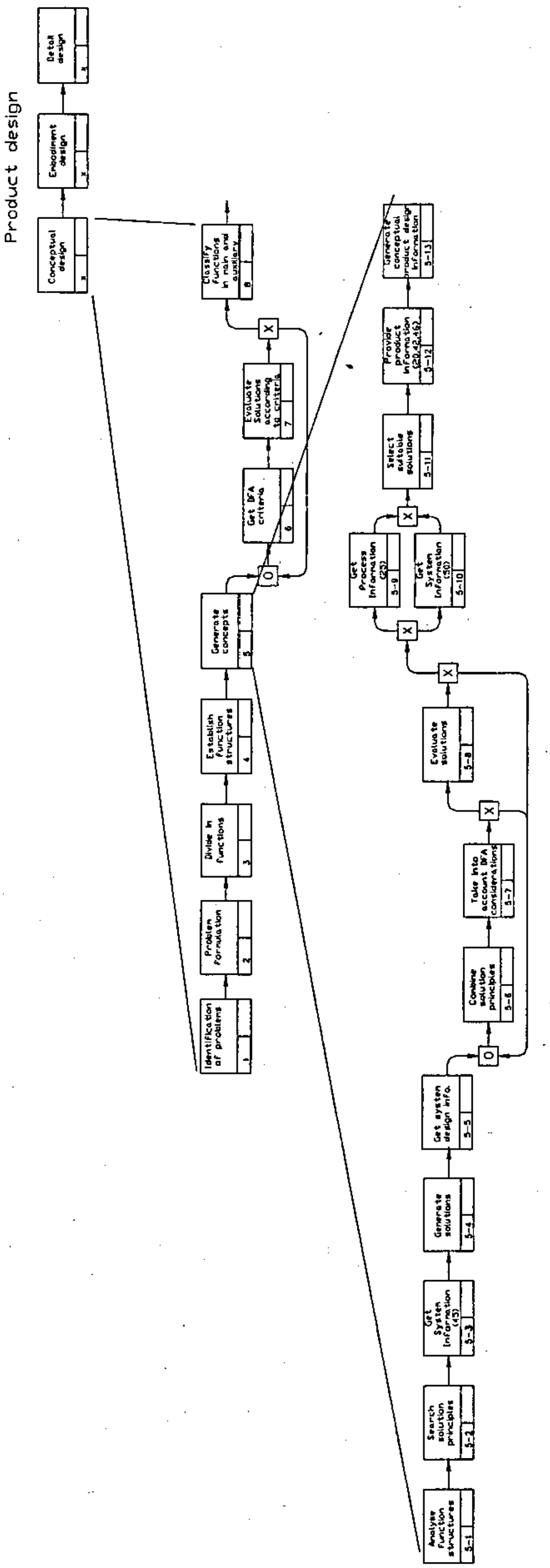



Figure D.1 IDEF3 Process centred description of the conceptual design stage of the product design

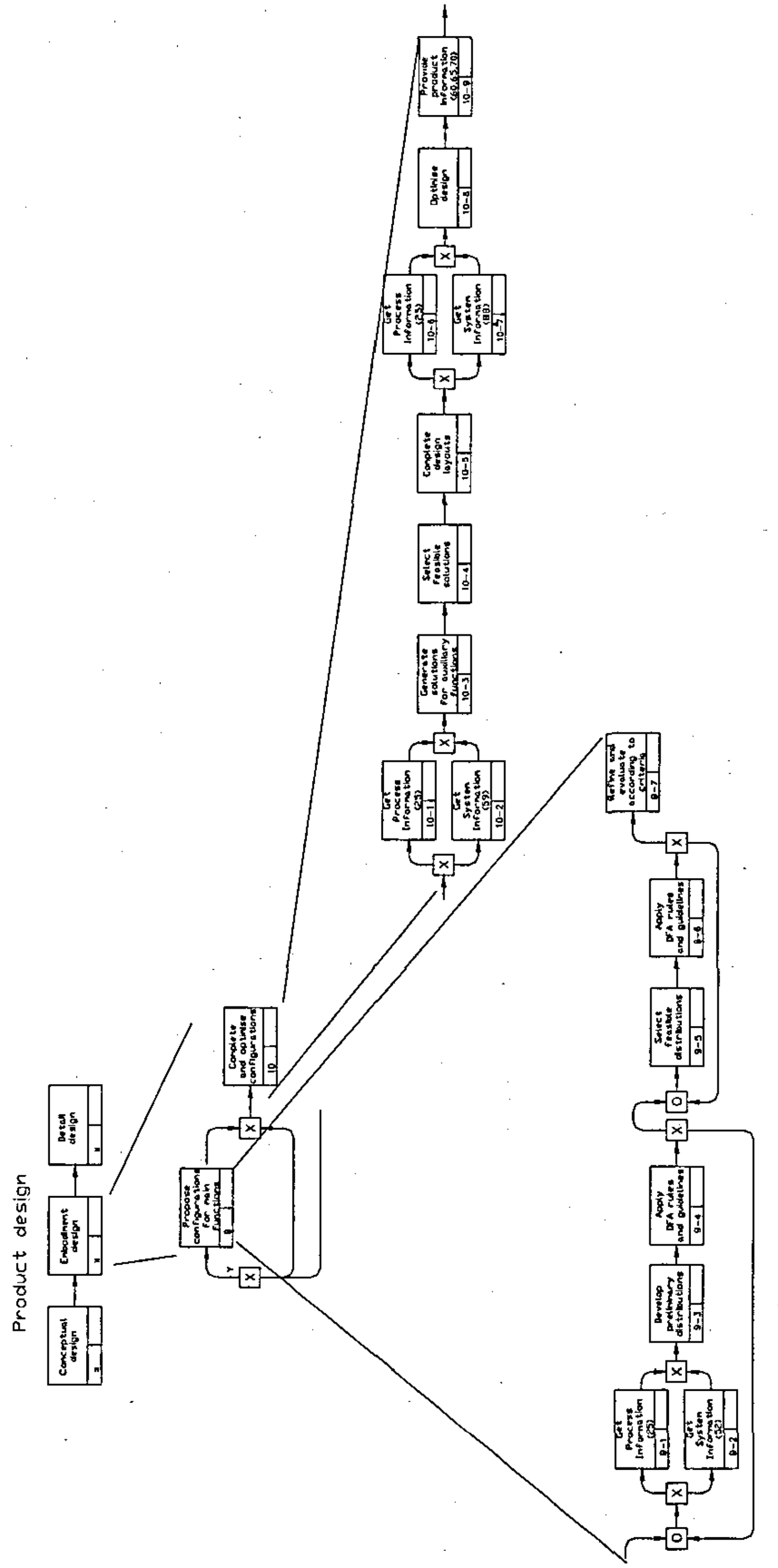



Figure D.2 IDEF3 Process centred description of the embodiment design stage of the product design

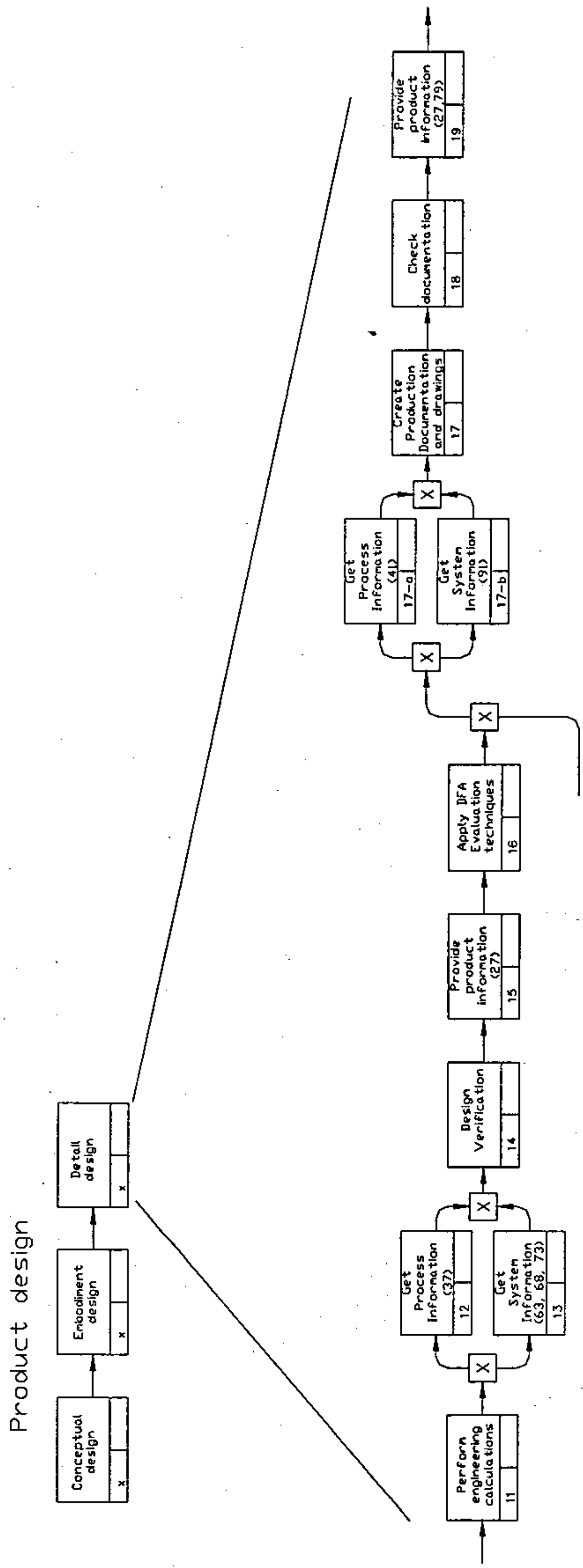



Figure D.3 IDEF3 Process centred description of the detail design stage of the product design

### D.1.1.2. Process Design

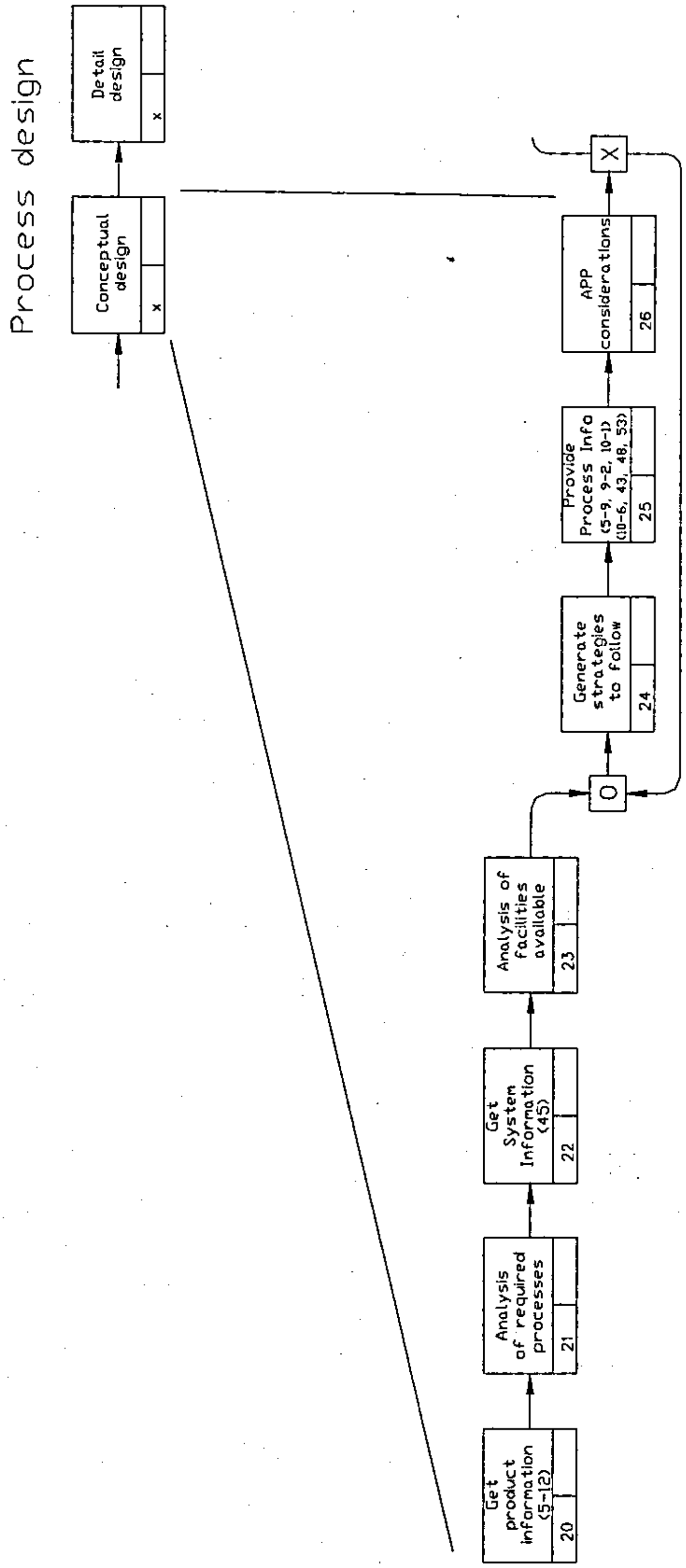



Figure D.4 IDEF3 Process centred description of the conceptual design stage of the process design

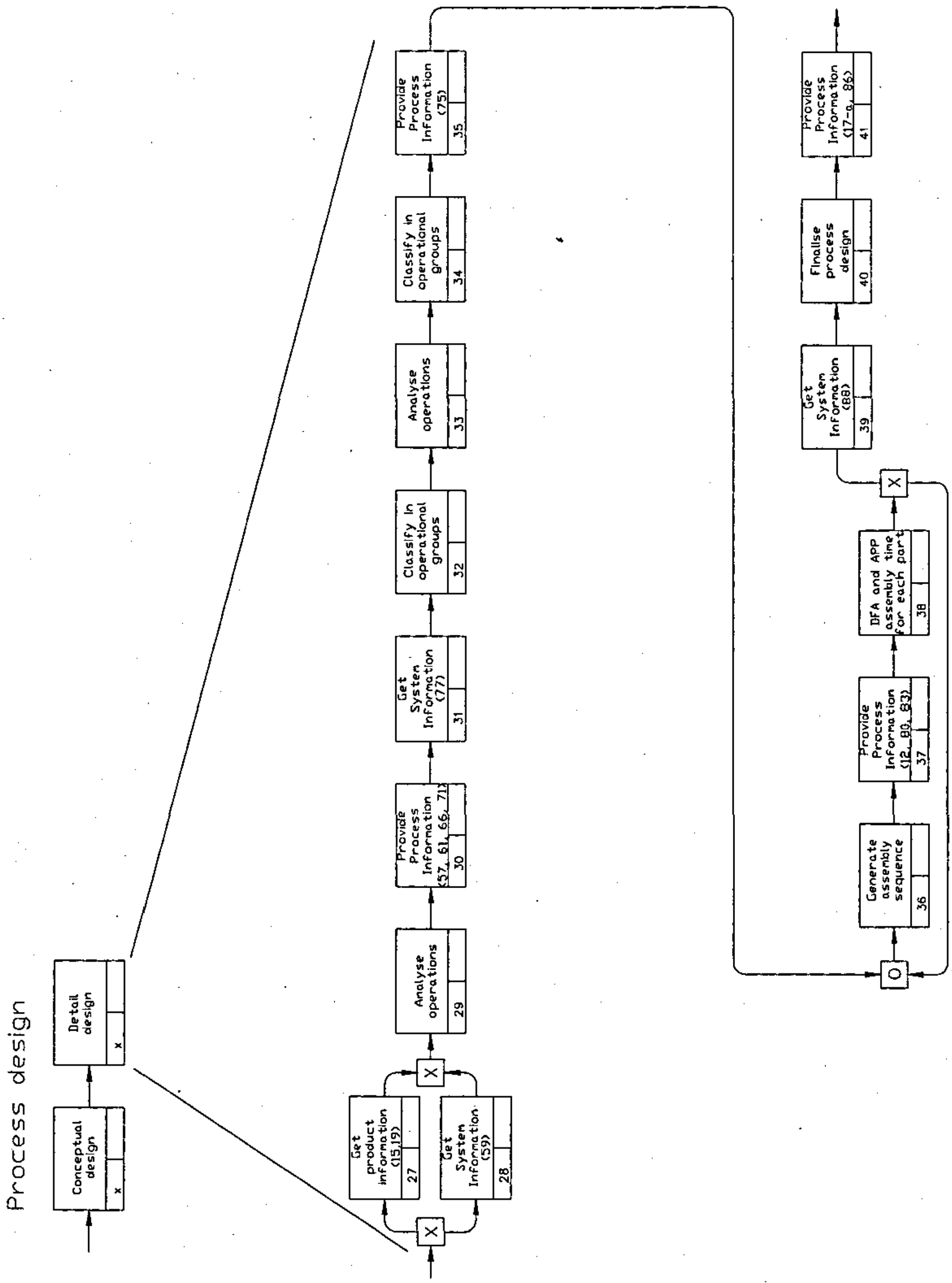



Figure D.5 IDEF3 Process centred description of the detail design stage of the process design

### *D.1.1.3. Resources Design*

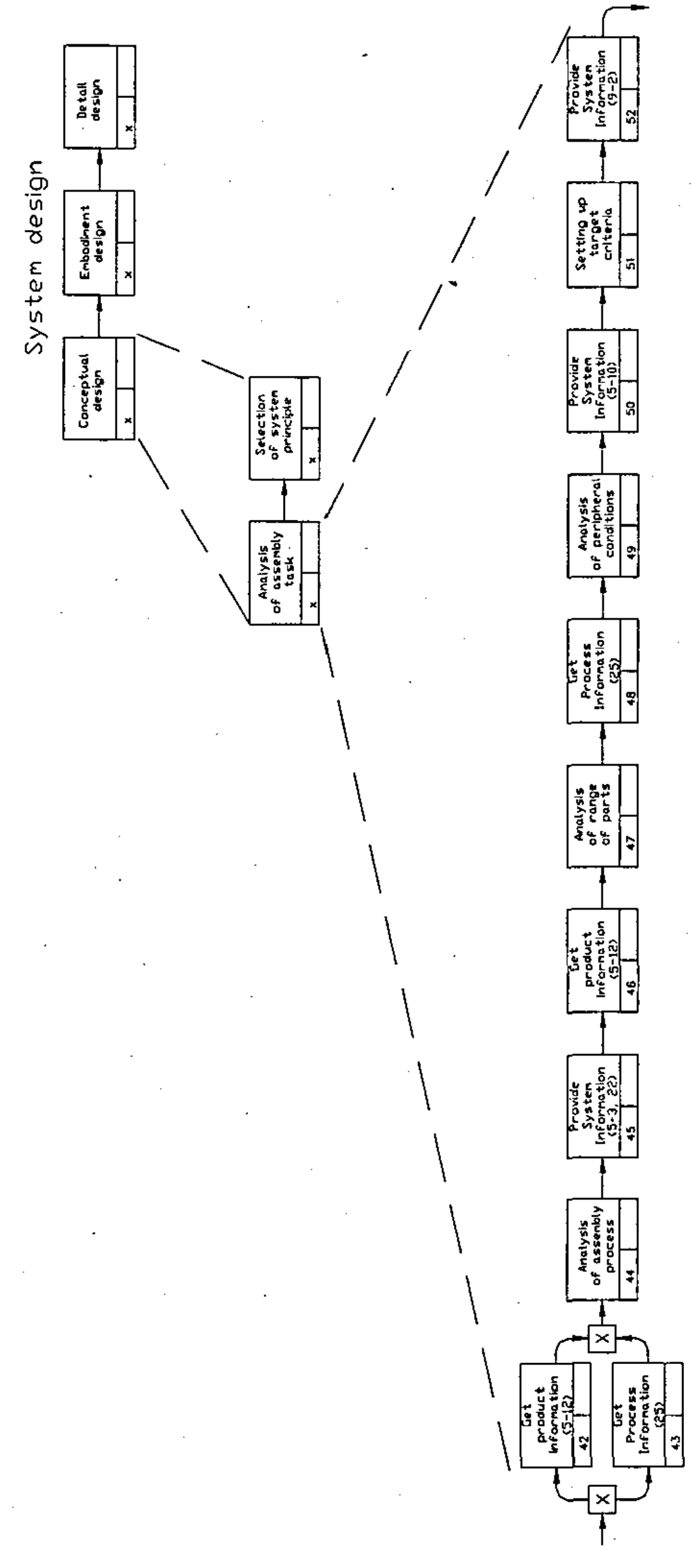



Figure D.6 IDEF3 Process centred description of the conceptual design stage of the resources design –analysis of the assembly

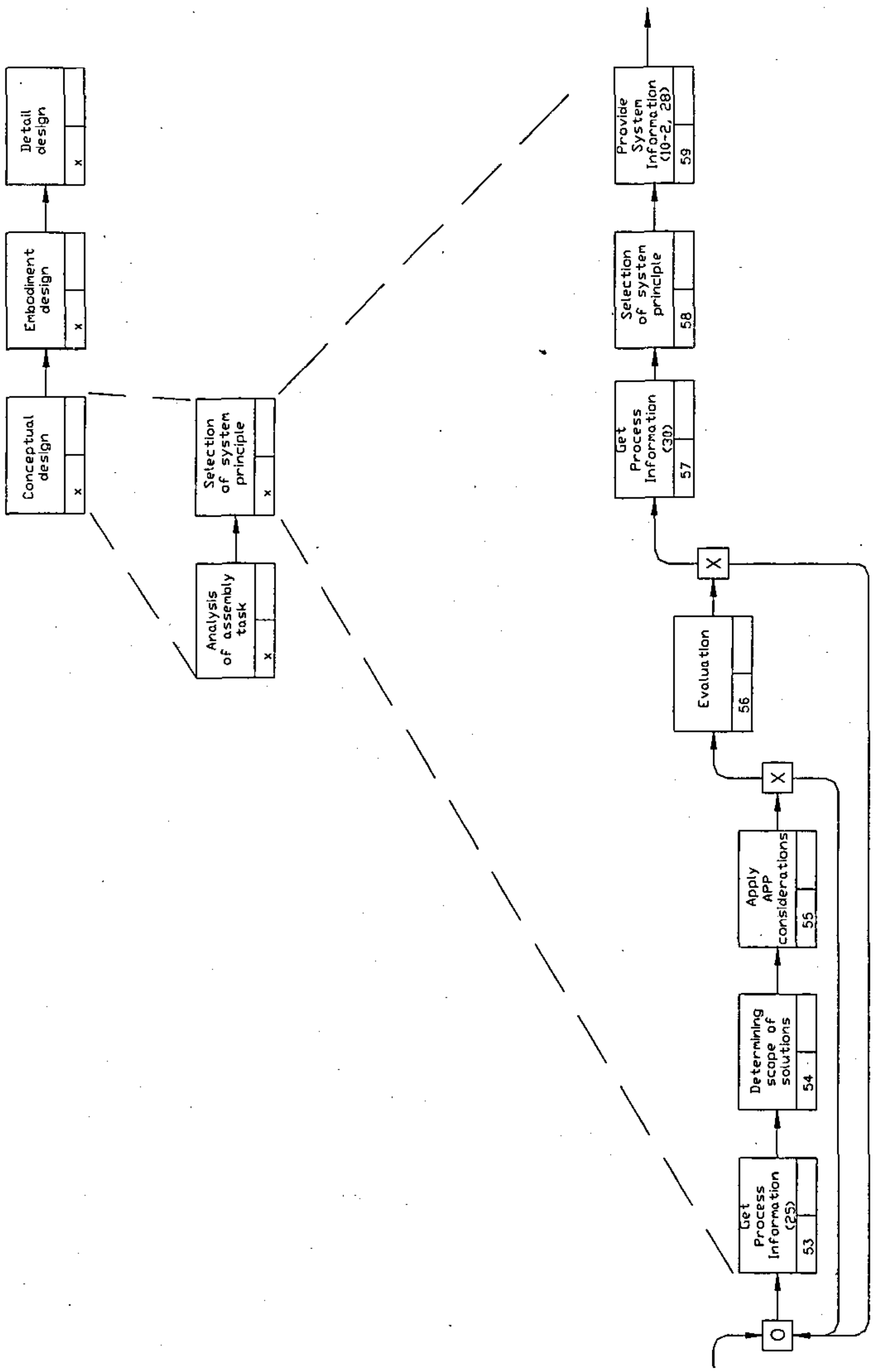



Figure D.7 Process centred description of the conceptual design stage of the resources design – selection of the system principle

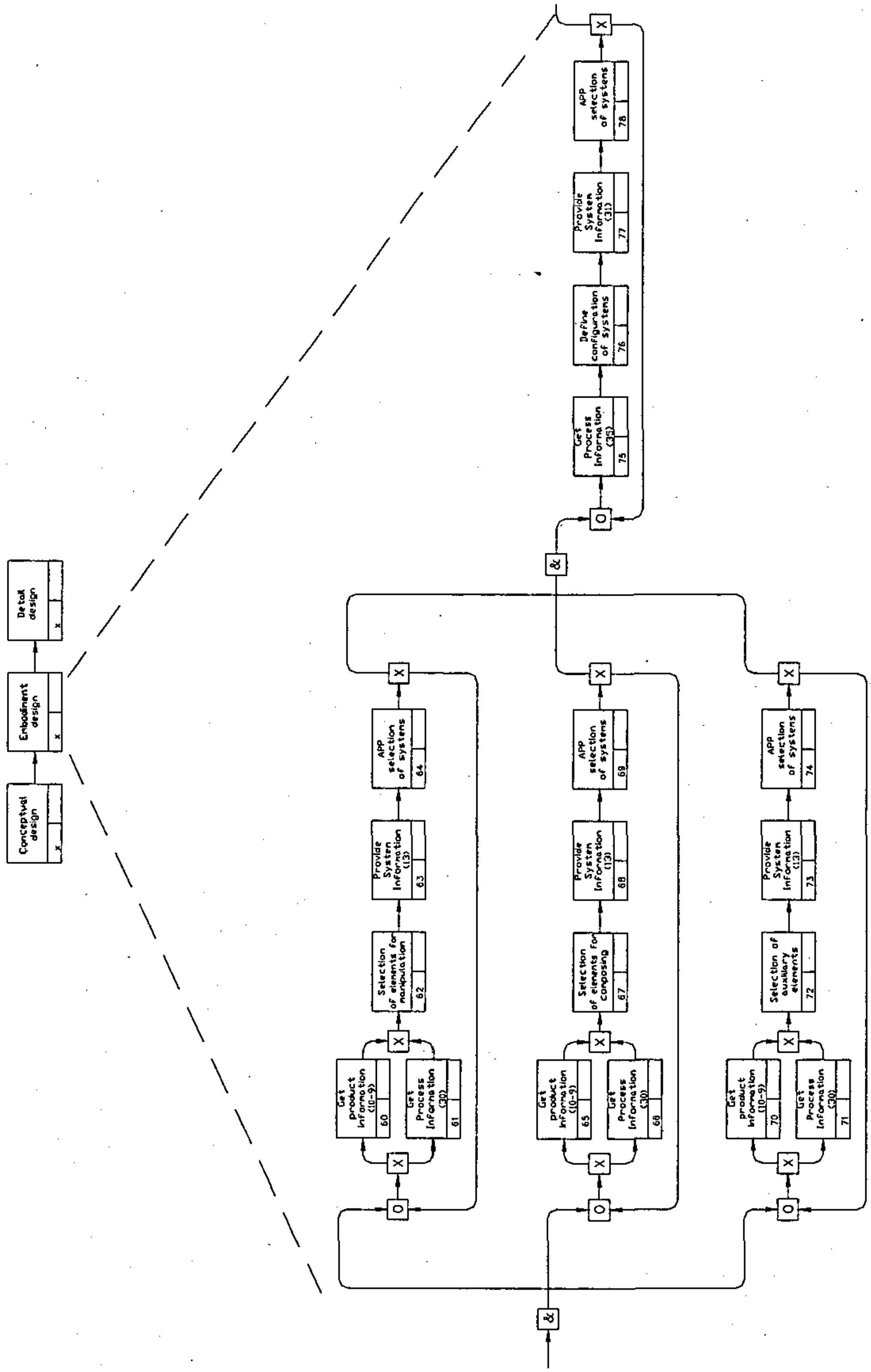



Figure D.8 IDEF3 Process centred description of the embodiment design stage of the resources design

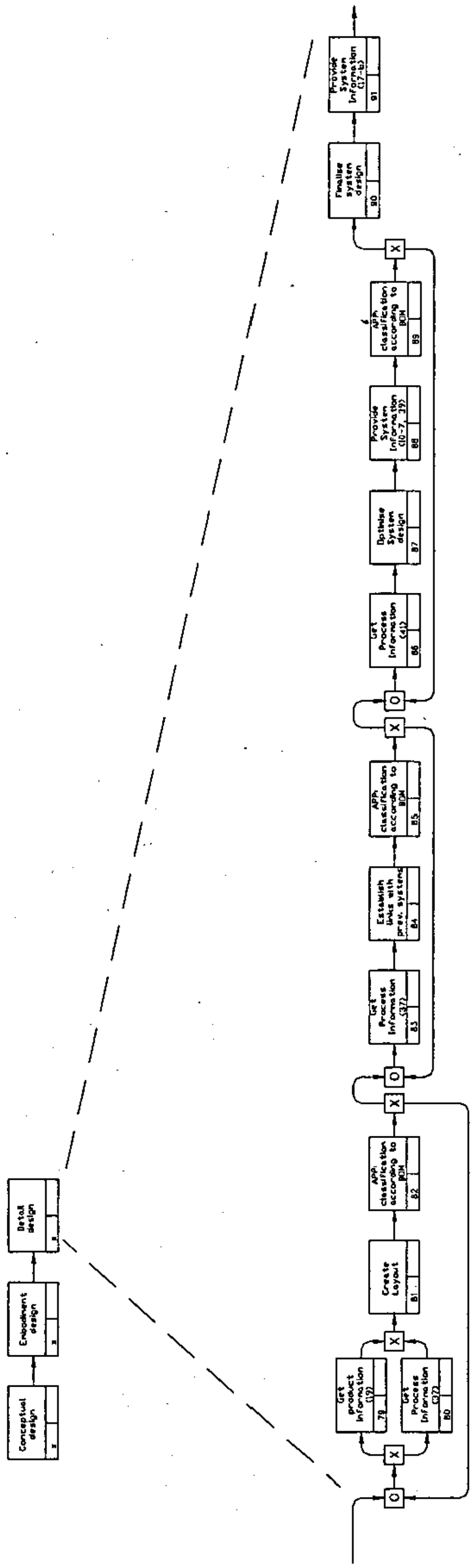


Figure D.9 IDEF3 Process centred description of the detail design stage of the resources design

### D.1.2. IDEF3 Object Centred Description Of The Assembly Activities

The IDEF3 object-centred description diagrams included in this section highlight the influence of DFA and APP through the concurrent design of the product, process and resources.

#### *D.1.2.1. Product Design*

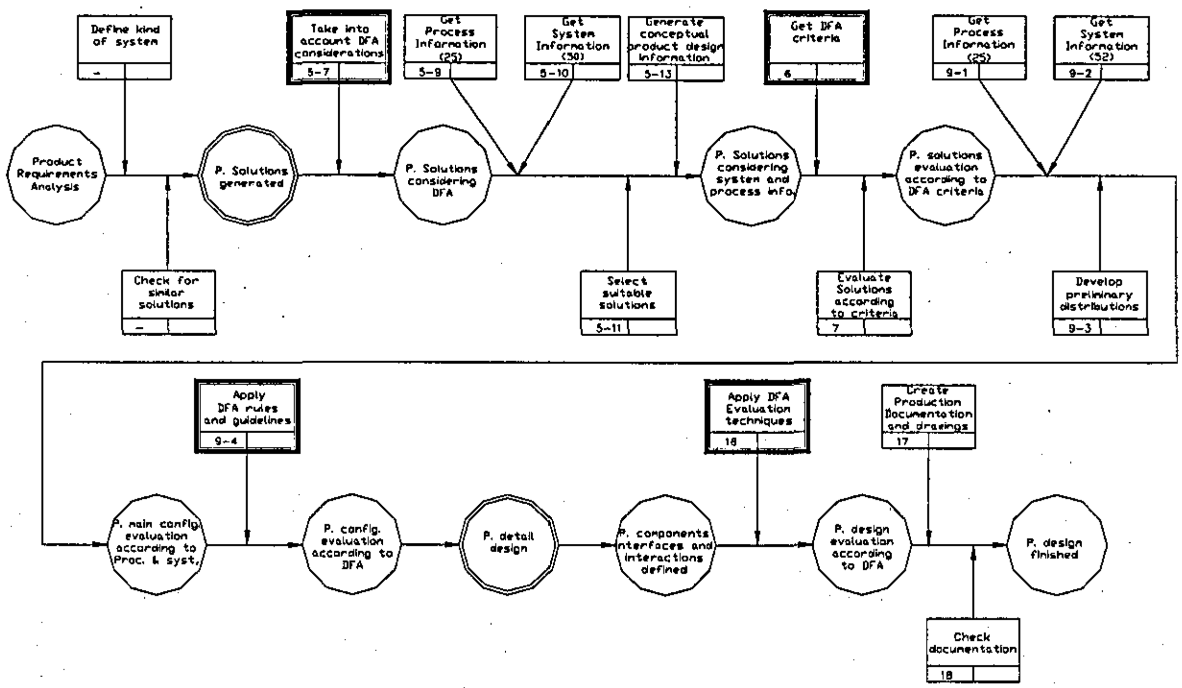



Figure D.10 Object centred description for the product design showing the interactions with DFA

### D.1.2.2. Process Design

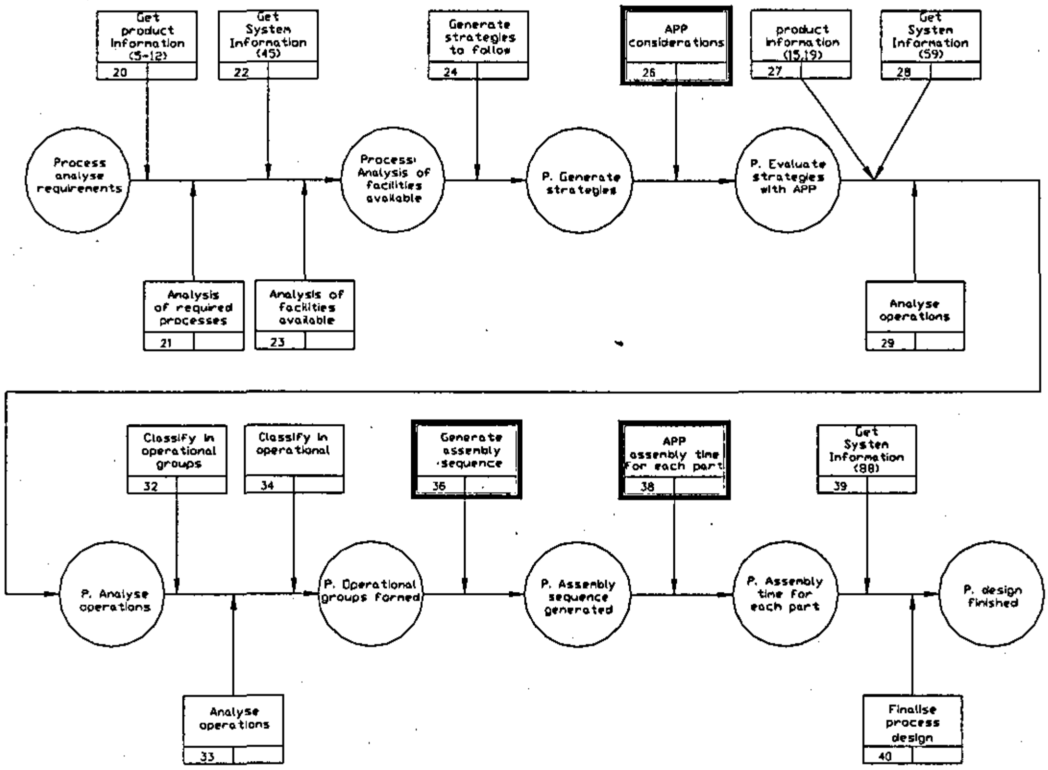



Figure D.11 Object centred description for the process design showing the interactions with APP

### D.1.2.3. RESOURCES DESIGN

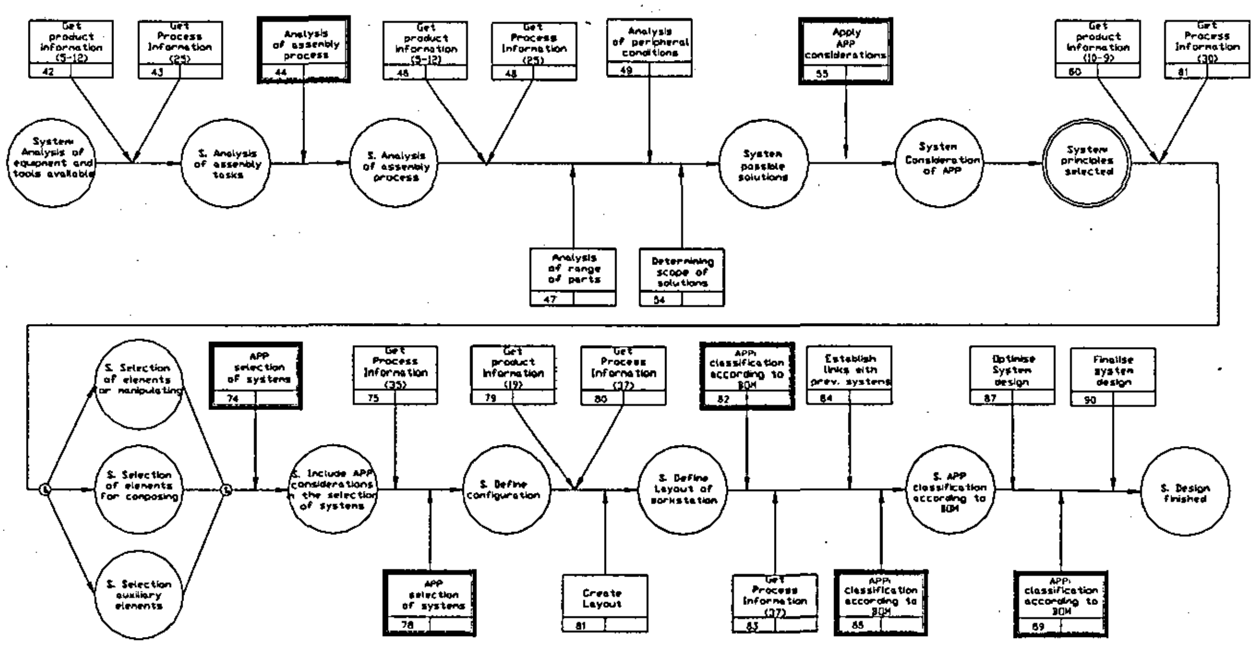


Figure D.12 Object centred description for the resources design showing the interactions with APP

## D.2. IDEF3 Diagrams for the Application of Design For Assembly and Assembly Process Planning through the Concurrent Design Process

### D.2.1. IDEF3 Process Centred Description Diagrams

The IDEF3 process centred description decomposition diagrams for the units of behaviour highlighted in section D.1.2 were used in Chapter 6 for obtaining the information requirements that DFA and APP have during the concurrent design of the product, process and resources.

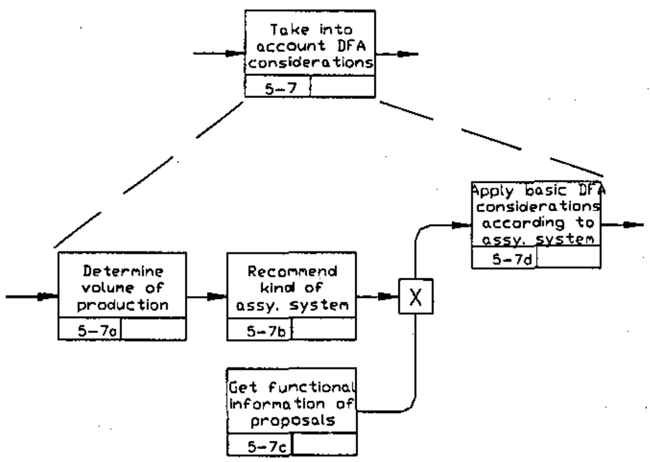



Figure D.13 Take into account DFA considerations in the conceptual stage of the design of the product

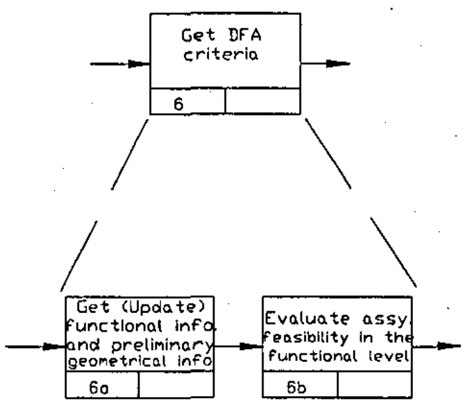



Figure D.14 Get DFA criteria and analyse the generated concepts

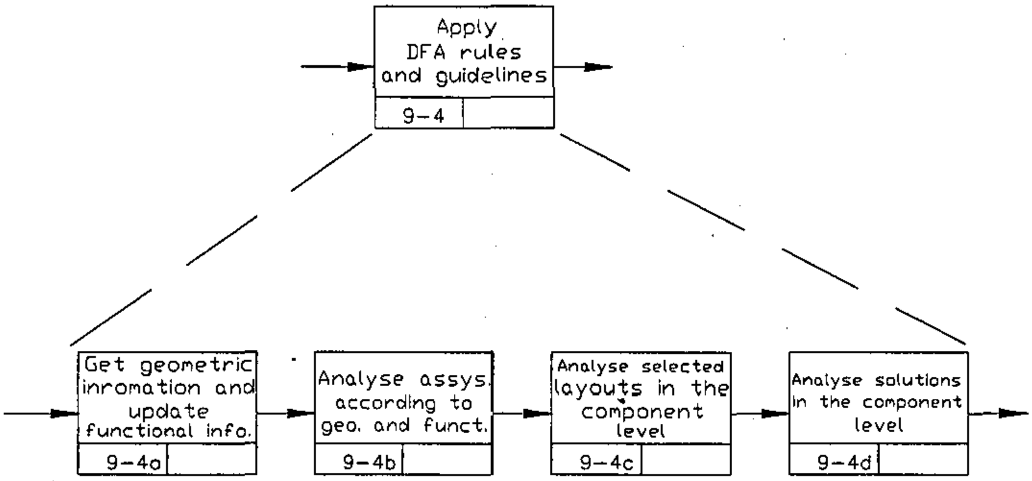



Figure D.15 Apply DFA rules and guidelines in the embodiment design stage.

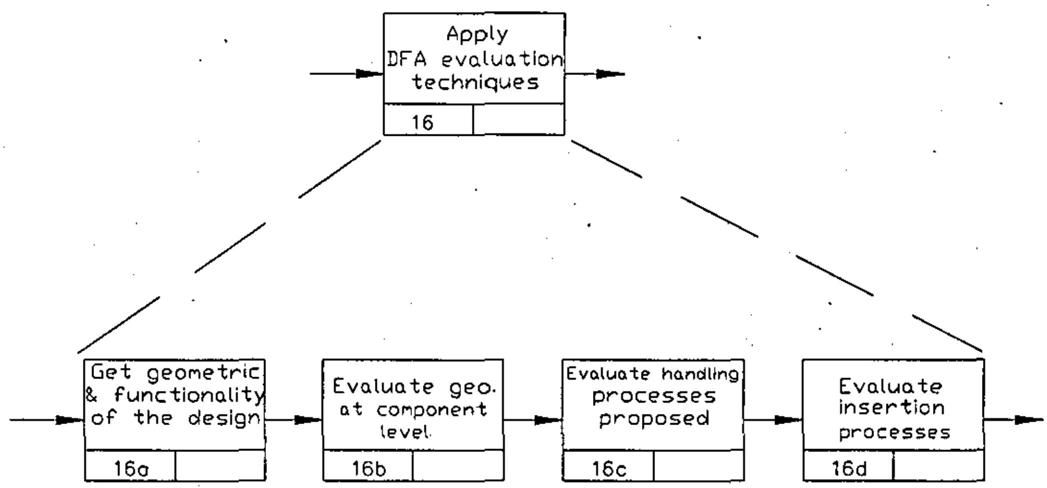



Figure D.16 Application of DFA evaluation techniques

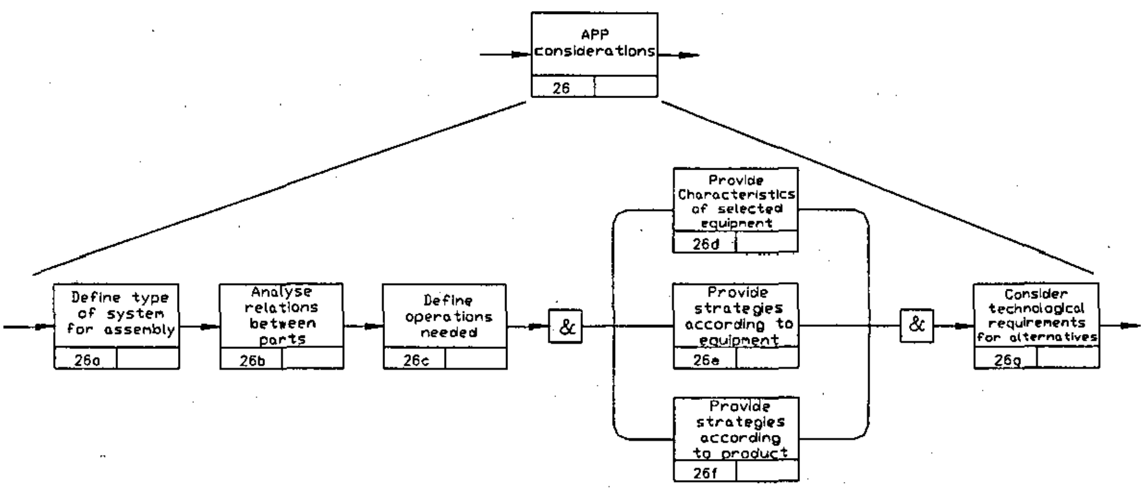



Figure D.17 APP considerations in the conceptual design of the process

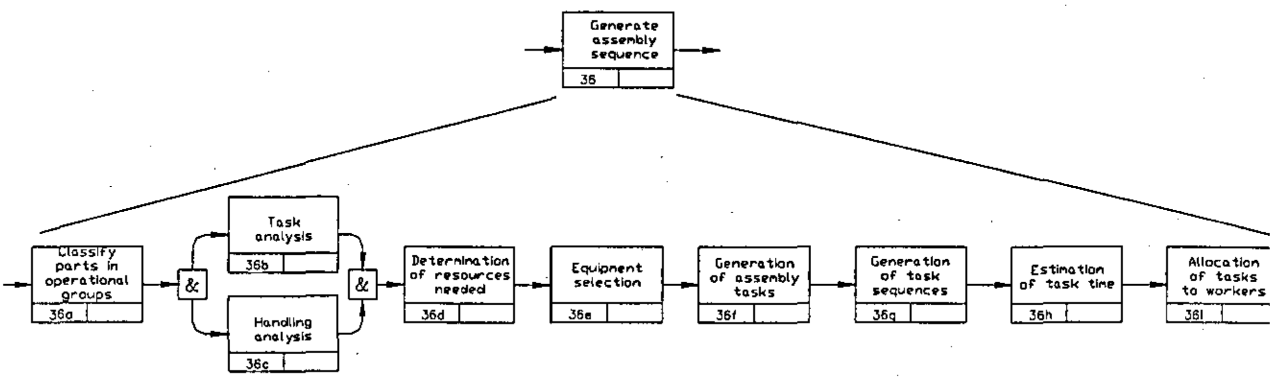



Figure D.18 APP generate assembly sequence in the process design

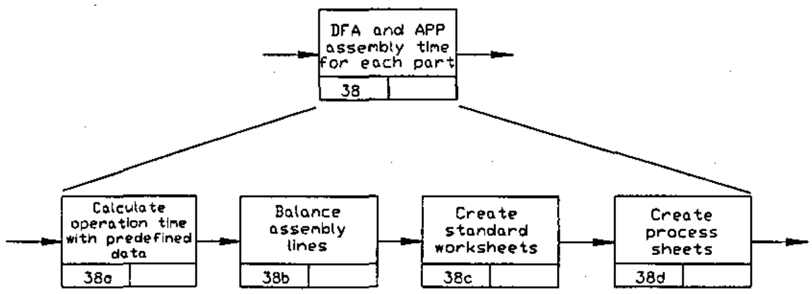



Figure D.19 DFA and APP assembly time definition for each process

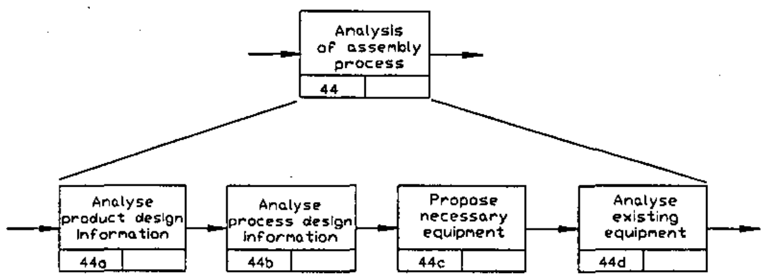



Figure D.20 APP analysis of assembly process in the conceptual design of resources

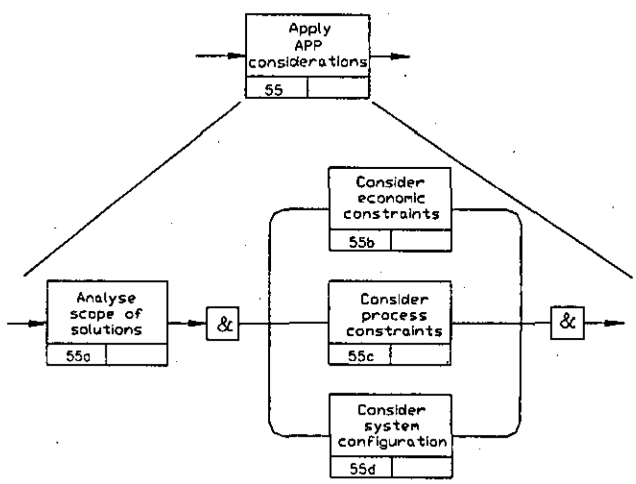



Figure D.21 Apply APP considerations in the conceptual design of the resources

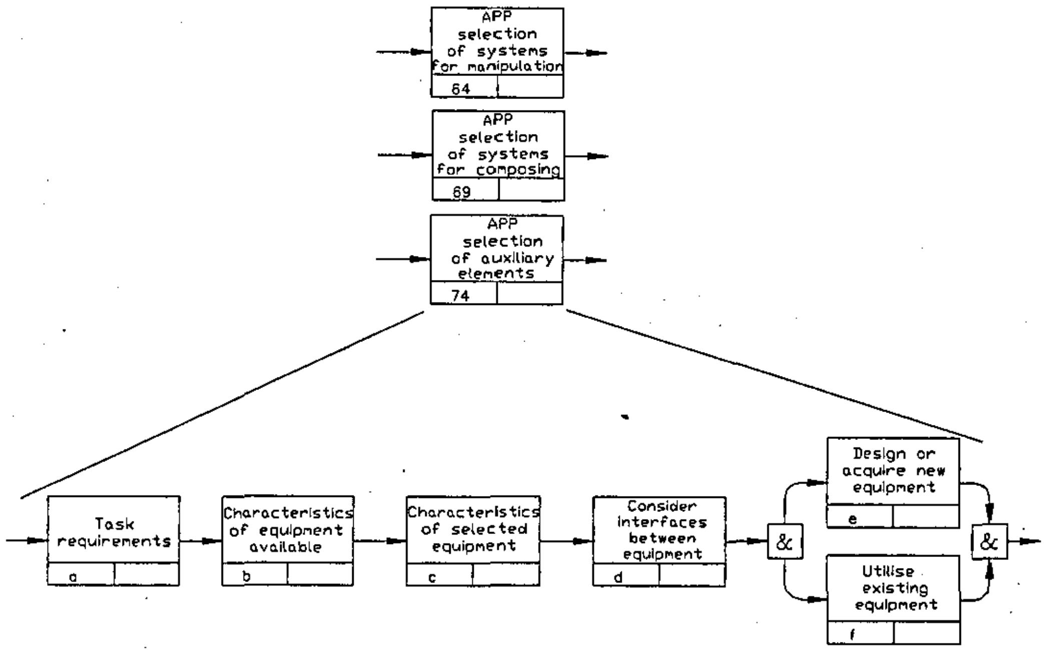



Figure D.22 APP support for the selection of systems in the embodiment design of resources

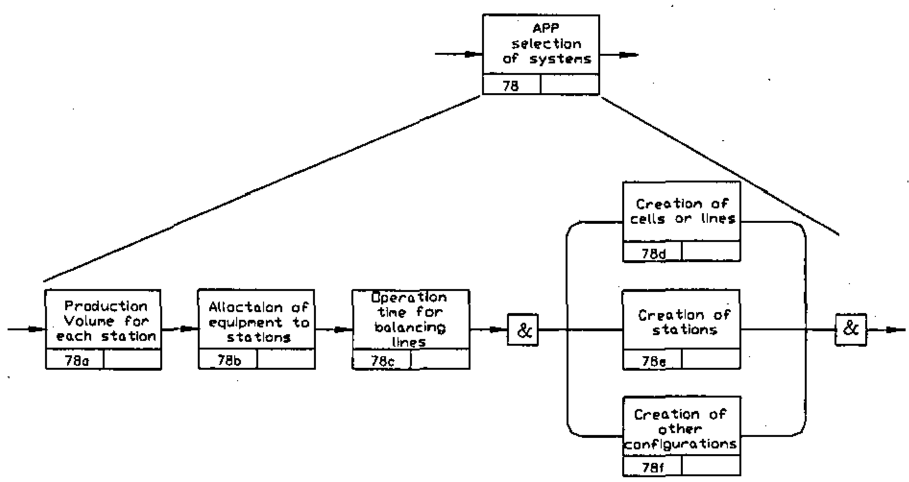



Figure D.23 APP selection of systems for the configuration definition of resources

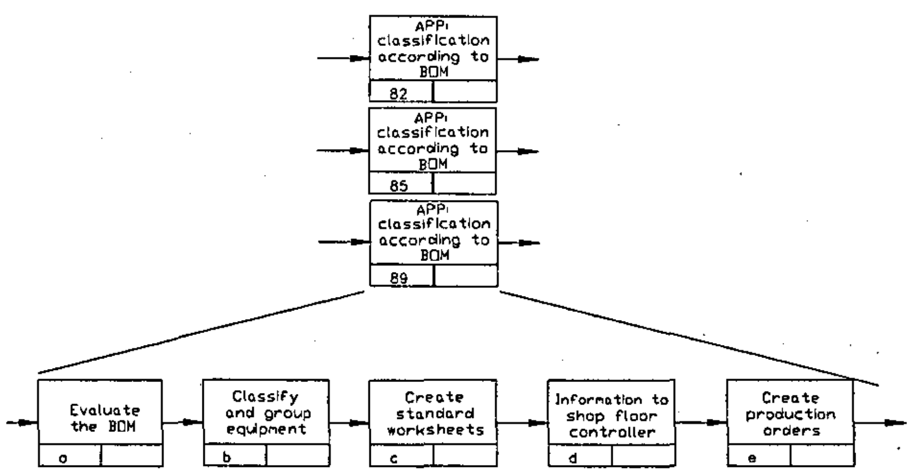



Figure D.24 APP classification according to Bill of Materials

## D.2.2. IDEF3 Unit Of Behaviour Elaboration Diagrams

This section includes the Elaboration Diagrams for the Units Of Behaviour included in the IDEF3 diagrams presented in section D.2.1, from Figure D.15 to Figure D.17

| | | | | |
|---|---|---|---|---|
| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | WORKING | REVIEWER | DATE: |
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | DRAFT | | |
| | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X RELEASED | | |

UOB No. 5-7a

**UOB Name:** **DETERMINE VOLUME OF PRODUCTION**

**Objects:** System User, statistics of current products, general characteristics of the product, estimates from marketing departments

**Facts:** The expected volume of production has to be determined in order to be able to define the kind of assembly system to be used. This estimation is done according to information from the marketing department of the company and from statistics of current products.

**Constraints:**

**Description:** The user has to introduce the estimated production volume to the system.

UOB No. 5-7b

**UOB Name:** **RECOMMEND KIND OF ASSEMBLY SYSTEM**

**Objects:** System User, Facilities Information, General characteristics of the product, table of recommendation for kind of system depending on product's characteristics.

**Facts:** The volume per hour, life of product, and number of parts determine the kind of system that can be used for assembling. The types of system are Assembly Station, Modular Assembly, Custom Assembly, Multistation Assembly, Assembly Cell, Dedicated Assembly Line, Dedicated Automatic Assembly Machine.

**Constraints:**

**Description:** The system shall evaluate the general characteristics of the product and recommend the most appropriate kind of system for assembly.

| | | |
|---|---|---|
| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |

| | | | | |
|---|---|---|---|---|
| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | WORKING | REVIEWER | DATE: |
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | DRAFT | | |
| | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X RELEASED | | |

UOB No. 6a

**UOB Name:** **GET (UPDATE) FUNCTIONAL INFORMATION AND PRELIMINARY GEOMETRICAL INFORMATION**

**Objects:** System user, functional information of the alternatives selected in the conceptual stage of product design.

**Facts:** At this stage of development, the product only consists on a series of ideas and concepts that comply with the functional requirements established for the product.

**Constraints:** The information about the conceptual product design has to be considered, even though the information is not complete.

**Description:** The geometric information obtained of the conceptual product design has to be retrieved in order to apply DFA considerations and guidelines.

UOB No. 6b

**UOB Name:** **EVALUATE ASSEMBLY FEASIBILITY IN THE FUNCTIONAL LEVEL**

**Objects:** System user, functional and geometric information of the product, DFA recommendations and guidelines

**Facts:** The system has to recognise geometric factors that affect assembly, such as symmetry, size and weight

**Constraints:** The information of the product can be incomplete or inaccurate, the analysis has to be done with the available information.

**Description:** At this stage of the product development, the geometric information of the options selected in the conceptual design are evaluated using DFA guidelines.

| | | |
|---|---|---|
| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

**UOB No. 9-4a**

**UOB Name:** GET GEOMETRIC INFORMATION AND UPDATE FUNCTIONAL INFORMATION

**Objects:** System user, functional and geometric information of the selected conceptual design of the product, requirements for the product

**Facts:** The geometric and functional information are key factors for applying DFA rules and guidelines

**Constraints:** The information being used will be constantly updated and modified, ensure to have the latest version for the analysis.

**Description:** All the available characteristics of the product have to be retrieved to perform the DFA analysis and give advice and guidelines to the user.

**UOB No. 9-4b**

**UOB Name:** ANALYSE ASSEMBLY ACCORDING TO GEOMETRY AND FUNCTIONALITY

**Objects:** System user, Product Geometry and Functionality, Assembly guidelines

**Facts:** The analysis has to be performed at the component and assembly levels.
The assembly guidelines are applied to find components that could have a problem in assembly, such as tangling, nesting, incorrect assembly, visibility problems, etc.

**Constraints:**

**Description:** The assembly of the product has to be analysed according to the geometry and functionality defined.

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

**UOB No. 9-4c**

**UOB Name:** ANALYSE SELECTED LAYOUTS IN THE COMPONENT LEVEL

**Objects:** System user, Product Geometry and Functionality, Assembly guidelines

**Facts:** The DFA analysis at the component level is focused on the geometric characteristics Size, Weight, Geometry, and Shape.

**Constraints:**

**Description:** The guidelines are applied by the system to the selected layouts in order to find the approaches that contain less problems or restrictions to the assembly, such as tight tolerances or surface finishes.

**UOB No. 9-4d**

**UOB Name:** ANALYSE SOLUTIONS IN THE COMPONENT LEVEL

**Objects:** System user, Product Geometry and Functionality, Assembly guidelines

**Facts:** The DFA analysis at the component level is focused on the geometric characteristics Size, Weight, Geometry, and Shape.

**Constraints:**

**Description:** The guidelines are applied by the system to the geometric characteristics existent in the Product Model, in order to advise the user of necessary changes at this stage to avoid problems in assembly.

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

| | | |
|---|---|---|
| UOB No. 16a | **UOB Name:** | **GET GEOMETRIC AND FUNCTIONALITY OF THE DESIGN** |
| | **Objects:** | System user, Product Geometry and Functionality, Assembly evaluation tables and rules |
| | **Facts:** | The information of the product is detailed, so the analysis can be performed using the geometric and functionality information of the product |
| | **Constraints:** | |
| | **Description:** | The geometric and functionality information of the product has to be retrieved in order to perform the assemblability analysis. |
| UOB No. 16b | **UOB Name:** | **EVALUATE GEOMETRY AT COMPONENT LEVEL** |
| | **Objects:** | System user, Product Geometry and Functionality, Assembly evaluation tables and rules |
| | **Facts:** | The size, weight, symmetry, handling features, insertion features, shape and tolerances are used for evaluating the product and detect problems such as very heavy parts, tangle or nesting, sharp or delicate parts. |
| | **Constraints:** | The possibility of eliminating the parts has to be evaluated |
| | **Description:** | All the characteristics of the part have to be evaluated to find possible problems in the assembly and they have to be solved by modifying the design of the product. |

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

| | | |
|---|---|---|
| UOB No. 16c | **UOB Name:** | **EVALUATE HANDLING PROCESSES PROPOSED** |
| | **Objects:** | System user, Product Geometry and Functionality, Handling evaluation tables and rules |
| | **Facts:** | The handling processes applied for the part have to be evaluated according to experience or tables and rules. |
| | **Constraints:** | Boothroyd, Lucas, or similar tables can be used. |
| | **Description:** | The manipulation (or automatic handling) of the product has to be evaluated using criteria included in tables and rules for handling, providing recommendations for solving such problems. |
| UOB No. 16d | **UOB Name:** | **EVALUATE INSERTION PROCESSES** |
| | **Objects:** | System user, Product Geometry and Functionality, Insertion evaluation tables and rules |
| | **Facts:** | The insertion processes applied for the part have to be evaluated according to experience or tables and rules. |
| | **Constraints:** | Boothroyd, Lucas, or similar tables can be used. |
| | **Description:** | The manual or automatic insertion of the parts has to be evaluated using criteria included in tables and rules for insertion, providing recommendations for solving such problems. |

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 PROJECT Product and Process Information Interactions in Assembly Decision Support Systems NOTES 1 2 3 4 5 6 7 8 9 REV: | WORKING / DRAFT / RECOMMENDED / X RELEASED | REVIEWER | DATE: |
|---|---|---|---|---|

**UOB No. 26a**

**UOB Name:** DEFINE TYPE OF SYSTEM FOR ASSEMBLY

**Objects:** System user, product general characteristics, resources and processes available, criteria for selecting the kind of enterprise

**Facts:** By analysing the general characteristics, the optimum kind of system for assembling the part, according to economic criteria, can be given.

**Constraints:** Different subassemblies of the same product can have different optimum type of assembly system. The decision relies on the user.

**Description:** The general characteristics of the part are analysed to propose the best kind of system to assemble it, this system can be manual, automatic, semiautomatic, dedicated machines, etc.

**UOB No. 26b**

**UOB Name:** ANALYSE RELATIONS BETWEEN PARTS

**Objects:** System user, product characteristics information, relations between the parts.

**Facts:** The relations between parts have to be analysed in order to define the operations needed for the assembly of the parts.

**Constraints:**

**Description:** The analysis of the relations between parts is performed in order to find the assembly operations required and to be able to select the equipment to be used.

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J.M. Dorador DATE: Mar, 2001 PROJECT Product and Process Information Interactions in Assembly Decision Support Systems NOTES 1 2 3 4 5 6 7 8 9 REV: | WORKING / DRAFT / RECOMMENDED / X RELEASED | REVIEWER | DATE: |
|---|---|---|---|---|

**UOB No. 26c**

**UOB Name:** DEFINE OPERATIONS NEEDED

**Objects:** System user, Product Information, Processes required, Resources available, Relations between the parts

**Facts:** Define assembly operations for the product.

**Constraints:**

**Description:** The operations required for the assembly are defined in order to be able to select the equipment for assembling the product

**UOB No. 26d**

**UOB Name:** PROVIDE CHARACTERISTICS OF SELECTED EQUIPMENT

**Objects:** System user, Information about the Resources available in the Facilities selected, information about the Product characteristics.

**Facts:** The characteristics of the equipment selected are evaluated in order to be able to select alternatives

**Constraints:** The capacity of the equipment has to be compared with the characteristics of the components.

**Description:** The characteristics of the equipment are retrieved.

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J. M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

| | | |
|---|---|---|
| UOB No. 26e | UOB Name: | PROVIDE CHARACTERISTICS OF SELECTED EQUIPMENT |
| | Objects: | System user, Information about the Strategies that define the use of Resources and Processes available in the Facilities selected, information about the Product characteristics. |
| | Facts: | The strategies and methods for the use of the equipment selected are evaluated |
| | Constraints: | |
| | Description: | The strategies for the use of the equipment are retrieved |
| UOB No. 26f | UOB Name: | PROVIDE STRATEGIES ACCORDING TO PRODUCT |
| | Objects: | System user, Information about the Resources and Processes available in the Facilities selected, Information about the Product characteristics. |
| | Facts: | The product-dependant assembly strategies are defined. |
| | Constraints: | The product-dependant strategies require a link between Product and Manufacturing Information. |
| | Description: | The strategies for the assembly of the product, according to its characteristics and connection types between the parts are selected. |

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|

| USED AT: Loughborough University | ANALYST: J. M. Dorador DATE: Mar, 2001 | | WORKING | REVIEWER | DATE: |
|---|---|---|---|---|---|
| | PROJECT: Product and Process Information Interactions in Assembly Decision Support Systems | | DRAFT | | |
| | | | RECOMMENDED | | |
| | NOTES 1 2 3 4 5 6 7 8 9 REV: | X | RELEASED | | |

| | | |
|---|---|---|
| UOB No. 26g | UOB Name: | CONSIDER TECHNOLOGICAL REQUIREMENTS FOR ALTERNATIVES |
| | Objects: | System user, Information about the Resources, Processes, and Strategies available in the Facilities selected, information about the Product characteristics. |
| | Facts: | The technological requirements for alternatives are used for selecting the equipment to be used for assembly |
| | Constraints: | |
| | Description: | The characteristics of the possible equipment, strategies defined for the Facilities and Product-dependant strategies are used for selecting the equipment that will be used for assembling the product. |
| UOB No. | UOB Name: | |
| | Objects: | |
| | Facts: | |
| | Constraints: | |
| | Description: | |

| CONTEXT-SETTING REFERENCE: | ITEM DESCRIBED: | FORM TYPE UOB Elaboration: |
|---|---|---|