A Manufacturing Model to Support Data–Driven Applications for Design and Manufacture

by

Arturo Molina Gutiérrez

A Doctoral Thesis
Submitted in partial fulfilment of the Requirements
for the award of Doctor of Philosophy of the
Loughborough University of Technology
March 1995

DECLARATION

This is to certify that I am responsible for the work submitted in this thesis, that the original work is my own except as specified in acknowledgments or in footnotes, and that neither the thesis nor the original work contained therein has been submitted to this or any other institution for a higher degree.

ACKNOWLEDGMENTS

I always feel that this is one of the hardest parts when writing a thesis, there are so many people that you want to thank and you are always worried about missing someone. However this is a unique chance and I hope I will get it right.

I would like to thank my supervisor Prof. R. Bell for his guidance, encouragement and support. Especially for given me the opportunity to collaborate with the MOSES Research Project, an experience which has been proven to be most valuable and fruitful.

The author is also grateful to the LUT MOSES research group, Dr. R.M.I. Young, Miss. J. Harding and Mr. T.I.A. Ellis. To Dr. Young (Bob) my thanks to share with me his ideas about the Manufacturing Model concept, our useful discussions and his challenges while developing mine. My appreciation to Miss Harding (Jenny) for being the first to face the exciting world of Objectivity and therefore simplifying my task in the implementation of the Manufacturing Model. I am very grateful to Mr. Ellis (Tim) for all the work we have done together in the Manufacturing Model, especially in the development of the taxonomies and the creation of the manufacturing processes element of the model. I also appreciate his friendship and I wish him every success in his PhD.

I wish to thank the Mazak European Factory for their willingness to help me and conversations than ensued. I have learned a lot with my case studies.

Appreciation is extended to two international groups which I have been constantly in contact: the ISO TC184 SC4 WG8, especially to the former Project 2 Leader Mr. Bernhard Katzy, and to the IFAC/IFIP Task Force on Architectures for Enterprise Integration, especially to Dr. Peter Bernus, Dr. Francois Vernadat and Professor Theodore J. Williams.

I would like to thank to all my colleagues in the former L10 Lab, I believe that we have built a friendly environment for doing research. My special thanks go to three good friends Dr. Ahmed Al–Ashaab, who is now working in Mexico, and Mr. Shahin Ramihifard and Mr. Peter Coleman, I am going to miss our dinner sessions.

Many thanks also to all the Mexicans in Loughborough and Sutton Bonington, especially to the "Sunday Basketball Team", keep the good work.

I am very grateful to my sponsors: the Mexican Government (CONACyT), Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM, Campus Monterrey) and ORS Award Scheme.

To my parents, mamá grande, my sisters and their families for all their love and support. Thank you mom and dad for being there when I needed you most, you have been an example to follow. I also want to thank my mamá–suegra for always encouraging Silvia and me to continue our studies.

Silvia, I do not have enough words to thank you for being my companion and support during all these six years I have been doing research. If I have succeed is because of you.

To my beloved Mexico, I have promised myself to do my best in order to help my country to face all the challenges we might encounter. Finally to God for allow me to be what I wanted to be.

DEDICATION

To Silvia "Mi Morcín"

and

To my Parents

Abstract

This thesis is primarily concerned with conceptual work on the Manufacturing Model. The Manufacturing Model is an information model which describes the manufacturing capability of an enterprise. To achieve general applicability, the model consists of the entities that are relevant and important for any type of manufacturing firm, namely: manufacturing resources (e.g. machines, tools, fixtures, machining cells, operators, etc.), manufacturing processes (e.g. injection moulding, machining processes, etc.) and manufacturing strategies (e.g. how these resources and processes are used and organized). The Manufacturing Model is a four level model based on a de–facto standard (i.e. Factory, Shop, Cell, Station) which represents the functionality of the manufacturing facility of any firm.

In the course of the research, the concept of data–driven applications has emerged in response to the need of integrated and flexible computer environments for the support of design and manufacturing activities. These data–driven applications require the use of different information models to capture and represent the company's information and knowledge. One of these information models is the Manufacturing Model.

The value of this research work is highlighted by the use of two case studies, one related with the representation of a single machining station, and the other, the representation of a multi–cellular manufacturing facility of a high performance company.

Table of Contents

Chapter 1

Introduction

Increased competition in the global market place has forced companies to look for methods and tools which will improve the quality of their products, reduce the product development time cycle and reduce production costs. A growing importance has been given to create methodologies and computer technologies which enable powerful information systems to be designed, and then make it possible to be integrated effectively into the modern factory. In these information systems a key aspect is the representation of good and high quality data. This thesis contains a timely contribution to this important aspect of information systems i.e the Manufacturing Model.

The rate of change in recent years has been startling. CAD, CAM, CIM, Integrated Manufacturing Enterprises, computer based modelling techniques (e.g. Enterprise Modelling, Information Modelling), improved practices (e.g. Simultaneous Engineering, Product Life Cycle Design), new concepts (e.g. Fractal Factories, Virtual Manufacturing, Agile Manufacturing) all claim to be key to industry's success. Each of these technologies, practices and concepts are gradually being pulled together to create the total integration of all stages of manufacture. All of them share one fundamental requirement: the need for advanced information technologies to integrate and coordinate various life–cycle considerations during product development activities. Therefore, it is now imminent that modern computer technologies and related information processing systems will play a major role in future manufacturing enterprises.

This thesis addresses two major areas of research:

1. The definition of a Manufacturing Model concept which represents the manufacturing capability information of manufacturing facilities in order to support concurrent design of products.

2. The establishment of a formalism to develop the Manufacturing Model concept in order to pursue a systematic approach for its generation.

The research reported in this thesis is concerned with the definition and development of an information model called the Manufacturing Model. The Manufacturing Model describes and captures the information about the manufacturing situation of a company in terms of its manufacturing facility and capabilities at different levels of abstraction based on a de-facto standard (i.e. Factory, Shop, Cell, Station). To achieve general applicability, the model consists of the information entities that are relevant and important for any type of manufacturing firm, namely: manufacturing resources (e.g. machines, tools, fixtures, machining cells, operators, etc.), manufacturing processes (e.g. injection moulding, machining processes, etc.) and manufacturing strategies (e.g. how these resources are used and organized by a particular company). This model provides reliable manufacturing information for the support of concurrent product realization. As the Manufacturing Model represents the detailed manufacturing capability of an enterprise, and its current manufacturing status, this model supports the formulation of new and better manufacturing business strategies, facilitates the development of enterprise models and will be a useful source of information for real time production control applications.

This PhD thesis has contributed and collaborated with two main research projects currently pursue by the Simultaneous Engineering Group at LUT :

1. Model Oriented Simultaneous Engineering System (MOSES) which is being researched in collaboration with Leeds University under ACME funding. A key issue in this research is the development of two information models which support the different perspectives required in concurrent product realization e.g. design for manufacture. These information models are the Product Model and the Manufacturing Model.

2. Conceptual Modelling – Linking Business Strategies with Manufacturing Performance. This research project has the following objectives: (1) research the linkage between business strategy and manufacturing performance; (2) research a strategic manufacturing decision–aid for top management and (3) investigate the generation of factory performance requirements for middle management

Especially, in this research work, the collaboration with two researchers Mr. T.I.A. Ellis and Dr. R.I.M. Young has proven to be very helpful. Additionally to this collaboration, the interactions with the relevant ISO TC184 SC4 WG8 activities (MANDATE) and IFAC/IFIP Task Force on Enterprise Integration have been very valuable.

The relevant literature reviewed during this research is reported in Chapter 2. Competitive research is described in Chapter 3 to position this work into a context and allow the author to assess his contributions. The future applicability of the Manufacturing Models is discussed in Chapter 4 in order to highlight its potential. The context of this research and its relation with the MOSES research concept is presented in Chapter 5.

The search for a formalism for the Manufacturing Model development has been a major issue in this research. There is an established practice of using reference models in certain areas of software engineering. However, it is an unproven approach in the general field of CAE systems. The work reported in this thesis shows how the concept of reference models can be used and in particular how its partial use can be readily effective. The creation of this formalism is explained over six chapters. It starts in Chapter 6 where the issues and problems tackled by the author are outlined and the major contributions are described to reflect how the issues and problems have been resolved.

The author's work was influenced by decisions made in the MOSES research project, between the two Universities, in relation to the development of a partial multi–view reference model of the MOSES CAE system. However there was a strong difference of opinions on how to supply the requirements level of the reference model. The solution to this dilemma, which is reported in this thesis, was to construct a formalism based on bringing together work from the CIMOSA project, and the Reference Model for Open Distributed Processing (RM–ODP). This work is reported in Chapters 7, 8 and 9.

The formalism, which is argued in this thesis, combines the CIMOSA Requirements Definition Model, and the use of RM–ODP as it has been explored within the MOSES research project. In this thesis this combination is referred as the CAE Framework. Chapter 7 describes this hybrid CAE Framework.

The use of the Requirements Definition Model of CIMOSA to capture the enterprise requirements for CAE systems is explained in Chapter 8. Chapter 9 discusses the utilization of RM–ODP to define the MOSES CAE Reference Model in order to describe CAE system functionality, configuration and technology that is necessary to satisfy the requirements specified by the CIMOSA Requirements Definition Model. The discussion related to reference models ends in Chapter 10 where the justifications about the choice of the methodologies and tools of the MOSES CAE Reference Model is presented.

The pursue of a formalism for underpinning the Manufacturing Model research has lead the author to conceive a novel Manufacturing Information Modelling Methodology. The concepts of this comprehensive methodology are described in Chapter 11. The use of this integrated methodology to created the content and structure of the Manufacturing Model is reported in Chapter 12. The author considers that all of these chapters constitute the core argument of this thesis.

Efforts have been made to put this novel thinking into use. Chapters 13, 14 and 15 report these initial efforts to apply the Manufacturing Model to an industrial instance. Three case studies have been elaborated in collaboration with the Yamazaki company. The example has been centred on the Mazak European Factory at Worcester. Details of the instances of the model and discussions with the company on these issues are contained in Appendices G, H and I.

Initially a Rotational Parts Station and related contemporary Turning Centre were taken as the focus of the work exploring the benefits of the Station Level of the Manufacturing Model (Chapter 13). Further work then was carried out to face the challenges of representing the operational rules that govern the use of a Machining Cell, and the possibilities of modelling the assembly process. Therefore it was required to study in detail the production planning and control mechanisms of the Mazak Factory, the Rotational Parts Line operation and the functionality of the assembly line. Chapter 14 describes the background information about these particular aspects of the Mazak Factory. The models developed to represent the Rotational Parts Line, related operational rules, and the assembly line are presented in Chapter 15.

This thesis contains a wide ranging set of issues and therefore a concluding discussion is presented in Chapter 16, which offers a final review of the balance of opinion on a number of major issues which then lead to the principal conclusions offered in Chapter 17.

Chapter 2

Literature Survey

2.1. Introduction

This literature survey aims to give an overview of how computer based information systems, engineering practices and related modelling techniques have been introduced in manufacturing companies to support the integration of design and manufacturing activities.

2.2. Contemporary CAD/CAM Technologies

Computer aided design and manufacturing (CAD/CAM) systems have evolved over the past four decades. The origins of Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM) can be traced back to when aerospace companies in both Europe and the USA began to experiment with computer based drafting systems (Taylor 1990). Around about the same time machine tool controllers were being developed to allow the pre-programming of machines by magnetic or paper tape methods (Kief and Waters 1992). Historically, there has been a separation between CAD, CAM and CNC because of the different skills involved. CAD is primarily a geometry-based activity and has developed very much with the needs of designers in mind. Because of this, CAM software from traditional CAD vendors has tended to be less well developed than the geometry related software (McMahon and Browne 1993). Those companies keen to exploit manufacturing automation have tended to adopt software written specially for manufacturing from CAM software vendors so, rather than having CAD/CAM and CNC, we now have CAD for modelling and CAM closely linked to CNC for manufacturing (Besant and Lui 1989).

2.2.1. Computer Aided Design (CAD)

CAD systems were created to support some of the activities in the human design process (Groover and Zimmers 1984). Bedworth et al. (1991) consider CAD as a complete set of computer tools which include all kind of aids for the designer, such as: drawing and drafting systems, wireframe, surface and solids modellers, Finite Element Analysis (FEA) packages, dynamic analysis and simulation packages, design advisors, etc. References related to the evolution of geometric modelling techniques in CAD systems i.e. drafting and drawing, wireframe, surface models, and solid models can be found in Spur et al. (1979), Requicha (1980), Requicha and Voelcker (1982), Mantyla (1988), Woodwark (1988), and Farin (1989).

McMahon and Browne (1993) argue that the connection or integration of the design and manufacture process is mainly influenced by the geometry representation and the technological description which has to be related to the geometry. Therefore this model is key to the successful automation of subsequent CAD/CAM integration procedures.

2.2.2. Computer Aided Manufacturing (CAM)

Computer Aided Manufacturing (CAM) can be defined as the effective use of computer technology in the planning, management, and control of the manufacturing function (Groover 1987).

Kidd (1992) has categorised the role of computers in manufacturing by the nature of the computer interface to the production process. Kidd (1992) proposes the following categories:

1. Indirect: in this case the computer's role is that of an information and decision support system, without any capability to directly sense the process (e.g. Computer Aided NC Programming).

2. Direct: in this case the computer itself directly monitors and controls sections of the manufacturing process (e.g. Direct Numerical Control–DNC, Computer Numerical Control–CNC, and Robotics).

Detailed studies of CAM evolution are described in Rembold et al. (1985), Groover (1987), Bruce (1990), Bedworth et al. (1991), Lynch (1992) and van Houten (1992).

2.2.3. Interfacing and Integrating CAD and CAM

McMahon and Browne (1993) define that the goal behind the total CAD/CAM system concept is the integration of all engineering operations, whether design or manufacturing oriented. Almost all information needed for a part definition is generated during the design stage. Nevertheless the problem of integration between CAD and CAM is primarily the representation and transformation of information among design and manufacturing e.g. the translation from geometric data to a set of manufacturing instructions and machining parameters. In Conkol (1990, 1991), the following conclusions have been drawn related to the current state of CAD/CAM system integration:

- CAD/CAM automation has impacted only drafting, process planning and NC programming to any significant degree.
- The current design environment does not support an adequate level of CAD/CAM integration.
- The current environments do not provide efficient communication between functional groups in an enterprise.

Eversheim et al. (1989) state that the CAD/CAM integration problem is related to the automation of both transmission and transformation of information between CAD and CAM. Main problem in traditional approaches is that generally the problem is looked at either from the design or manufacturing perspective (Shah and Wilson 1988). Therefore CAD/CAM systems do not provide the means to model the effects of a process in the design stage in order that the model can also be used in the automatic planning of machine operations. The problems are related to the heavy reliance of the model on the geometry rather than on the application.

Manufacturing information, which needs to be incorporated, is wide ranging from simple materials specifications to representation of tolerances, surface finish, and even process description (Grabowsky et al. 1989). The problem identified by Bey (1989) is that all these sort of information was not held in traditional CAD databases. As a result there is an inability to support the automatic querying and retrieval schemes, which in turn inhibited the development of automatic planning/programming in the machining, assembly, and other disciplines in manufacturing (Finger and Dixon 1989).

Eversheim et al. (1991) define that integration in CAD/CAM systems has to be at two levels:

1. Data/Information integration through the exchange of data/information between the CAD and CAM applications without losing intention, content, etc.

2. Functional integration through the communication between CAD and CAM applications which organizes and links the various functional areas of an enterprises to work together more effectively and optimize the whole enterprise.

The integration is much more powerful if the translation from CAD to CAM is more than just data/information transfer (Krause et al. 1989). Functional integration is important to allow for a much faster response time for prototyping, job changes, and job realization. Nowadays, the key to good CAD/CAM integration is well planned communication across functional areas in the enterprise, or even between enterprises. The capabilities of the CAD/CAM system have direct bearing on an efficient and effective communication.

Kimura (1992) concludes that research on new CAD/CAM systems has to consider not only technological aspects but organizational issues as well. Kimura (1992) defines two requirements that could make this possible:

1. The CAD/CAM system must be flexible enough in its capabilities to allow for a variety of information transfer. The system requires that all information about geometry (e.g. dimensions, tolerances, surface finish, definitions of surfaces and edges) and other parameters (e.g. material type, functionality), that appear on a

standard engineering drawing, be stored in a concise, organized format within a common data base.

2. The design and production departments (companies or engineers) must work together to devise new engineering practices to better use the CAD/CAM technology.

In order to cope with the first requirement different techniques have been developed to facilitate the information exchange between design and manufacturing. The following subsections review the evolution of integration techniques for CAD/CAM systems. New engineering practices (e.g. Simultaneous Engineering) have been introduced in order to tackle the second requirement, section 2.3 describes such practices.

2.2.3.1 Data Exchange using Neutral Formats

Davies (1991) defines three different methods for data transfer which have been used to achieve data/information integration in CAD/CAM systems:

1. Direct translators
2. Neutral format translators
3. Proprietary format translators

This literature review focuses on data exchange based on neutral formats which has been used commonly in industry to transfer data between CAD and CAM systems. The neutral format idea originated the development of a standard for product data exchange (Bey and Gengenbach 1988). Krause et al. (1989) argue that many of the issues surrounding standard data exchange arise because of incompatibilities between the various CAD systems .

The Initial Graphics Exchange Specification (IGES) was probably the most widely used neutral format in the world, particularly for mechanical engineering applications (CADDETC 1990). Using IGES it is possible to transfer three main classes of information: geometry, annotation, and structure. Applications such as draughting, wireframe, surface, electrical, finite element, constructive solid geometry and piping are covered by IGES (Bloor and Owen 1991).

Standard d'Echange et de Transfert (SET) developed in France (AFNOR 1989) covers draughting, wireframe, surface models, solid models, schematics, finite element, NC tool–paths and scientific data applications

Two standards created by the German automotive industry (VDAFS and VDAIS) have been used by the European automotive industry. The former VDAFS (Verband des Automobilindustries Flachen Schinttstelle) was developed to be limited in scope to polynomial curve and surface modelling. This standard has been replaced by VDAIS (Verbaud des Automobilindustri IGES Subset) version 2.0 which specifies seven subsets of IGES 4.0 and covers simple geometry, draughting and application data (Davis 1991).

Burkett and Yang (1992) conclude that five are the primary sources of error in data exchange using a neutral data exchange standard such as IGES, SET, and VDAIS:

- poorly written translators
- poorly designed/written standard
- ambiguities in the data exchange standard
- mismatches in semantics
- limited data exchange vocabulary

STEP (STandard for the Exchange of Product model data) was proposed by ISO TC184 SC4 in order to address the problems of previous data exchange standards and with the intention of supporting the complete representation of a product throughout its life cycle. The following objectives were formulated (ISO CD 10303–1):

- Use of Data Modelling
- Focuses on Multiple Application Domains

- Implementation Independence

Data Modelling was employed to address the ambiguities in the data exchange standard and mismatch in semantics. A significant difference between the development of IGES and STEP is the methodologies and techniques used in the development process (Shah and Mathew 1991). In STEP, technical committees developed reference models that defined the information needed to support a specific application. These reference models were data models that represented the type of information used by the application. These reference models would constitute the Resource Information Models of STEP, the formal definition of the information to be used in STEP implementation (Owen and Brett 1992). STEP is intended to address the data exchange needs of an extremely wide variety of subject areas. It was supposed to satisfy the product data communication needs of a product throughout it's life and across different industries. The obvious implication of this objective is difficulty of incorporating industry specific commonalities, and the standard has to be more general in order to handle the wider variety of requirements (Yang 1991).

STEP is organized into six logical groups or parts, each called a class. Each class has an unique function in the standardization of product data. The description of the classes is as follows (Bloor and Owen 1991).

1. Introductory: contains Part 1 of STEP, and provides an introduction to the concepts and fundamental principles of STEP and the structure of its parts.
2. Description methods: standardizes the methods used when describing STEP entities. EXPRESS modeling/programming language is used for describing the information models.
3. Resource Information Models: defines the data content that provides the basis of definition for product data. The product data is encapsulated in an implementation–independent form, and is only implemented via an application protocol.

4. Application Protocols: states explicitly the information needs of a particular application, specify an unambiguous means by which the information is to be exchanged for that application, and provide conformance requirements and test purposes for conformance testing.

5. Implementation forms: describes the multiple implementation methods that are supported by the logically complete information model for product data provided by STEP.

6. Conformance–testing methodologies: includes the definition of the standard procedures and tools required to undertake conformance testing of products that claim to implement one or more STEP application protocol standards.

The implementation independence and the entire STEP architecture (description methods, resource information models, application protocols, implementation forms and conformance–testing methodologies) were adopted to address all the sources of error identified above (ISO CD 10303–1). An independence between physical file format from the specification of the information content (e.g. resource information models and schemas) was intended to ease the development and maintenance of translators. Application Protocols (APs) address the limited vocabulary problem without a proliferation of entities at the resource information model level (Integrated Resources–IR). The differentiation between APs and IRs addresses the ambiguities in the standard and mismatch in semantics problems (Owen and Brett 1992).

2.2.3.2 Computer Aided Process Planning (CAPP)

Computer Aided Process Planning (CAPP) systems were considered the missing link between CAD and CAM applications (Chang and Wysk 1984, Allen 1987, Srihari and Greene 1988).

Detailed reviews of CAPP systems can be found in Steudel (1984), Eversheim and Schultz (1985), Ham and Lu (1988), Alting and Zhang (1989) and ElMaraghy (1993). This survey focuses on the role of CAPP system as the means to achieve integration of

CAD and CAM. Basically, the author has identified three different ways of how CAPP systems can interpret, structure and use design information to perform manufacturing process planning and generate NC–code:

- using Coding and Classification Systems (Group Technology)
- performing feature recognition and extraction
- using features to produce process plans and NC–code.

Ham and Lu (1988) identified that in earlier CAPP systems, the link between design and manufacturing information was provided through the use of Group Technology (GT). With GT the part characteristics and features are represented in the form of a code, which can be used as input to process planning systems. Nevertheless, Srihari and Greene (1988) recognised that interpretation of part characteristics has to be performed manually and exact size information can therefore be lost; hence, GT code alone is not suited to complete automation.

The link among the part model containing the geometry, topology, their relationships, and the application–specific spaces (like process planning and NC code generation) can be thought of as a set of features (Hummel and Brown 1989). A feature is any geometric form or entity whose presence or dimensions in a domain are applicable to manufacturing evaluation or planning, or to automation of functional analyses (Dixon 1986, Cunningham and Dixon 1988, Shah et al. 1989). Instead of a model consisting of graphic primitives (lines, points), a set of features (holes, slots, pockets) from which operations can be derived are defined (Luby et al. 1986, Unger and Ray 1988, Hummel and Brooks 1986). Features are sometimes thought of as volumes to be removed by machining operations (Woo 1984, Henderson and Anderson 1984, Faux 1986, Pratt 1988). Researchers have used features to refer to artifacts such as gears, bearings, shafts, cams, etc. (Krause et al. 1987, Murakami and Nakajima 1987).

The investigation for obtaining a part description which could completely integrate CAD/CAPP/CAM led to the identification and recognition of features from geometric representation (Dixon 1986). Feature recognition and extraction assumes that a product has been designed and that a geometric representation is stored in a solid modeling database using a specific format such as a data exchange standard (IGES, STEP, etc.) or a CAD data model representation (CSG, BRep, etc.). Original work in features extraction was proposed by Woo (1975, 1977) as a methodology for extracting geometric feature information from a CSG model for 2–1/2D parts. He focused on cavity recognition to transform volumetric designs of parts into descriptions for NC machining. Woo (1982) presented a novel technique for extracting features from a BRep file to yield a series of volumes with alternating signs for volume addition and subtraction called the Alternating Sum and Volume (ASV) expression. Woo (1984) extended the ASV concept to provide a data structure conversion between two modelling interfaces (CSG and BRep).

In the simplest 2D form, IGES can represent an engineering drawing. But items such as dimensions can be represented in various ways, and different drafting systems use different technologies to group lines into profiles. So there appears to be major problems in using IGES to transfer data between different systems. Work on features recognition and extraction from IGES representation has been undertaking by Li (1988), Park et al. (1990), and Muthsam and Mayer (1990).

In 3–dimensional (3–D) CAD, models provide another computer readable form of part description which can be used to extract knowledge about shape, size, surfaces, relationship between surfaces, and to drive a process planning system. In 3D the problems get worse because of the many incompatible ways of sorting surface and space curves. Some other attempts, such as Boundary Representation (BRep) and the Constructive Solid Geometry (CSG) tree do not provide any semantic information which could be associated with the machined volumes and are based on local information (Henderson 1986, Oh and Lee 1990).

Different researchers have worked on applying syntactic pattern recognition for extracting features from an edge list or line drawings (Jakubowski 1982, Kyprianou 1983, Staley et al. 1983, Choi et al. 1984, Liu and Srinivasan 1984, Srinivasan et al. 1985, Li 1988). Rules play an important role in feature recognition, which is essentially a process of finding groups of elements conforming to certain predefined patterns. The feature recognition algorithms of Henderson (1984), Kung (1984) and Joshi et al. (1989) are all based on this approach.

Alting and Zhang (1989) concluded that few CAPP systems have realized interfaces to CAD, CAM and some other computerized systems The following problems have been identified by van Houten et al. (1989) in the interfacing schemes which use feature recognition and extraction :

1. Difficulty of including process semantics when features are identified.
2. Limited scope of feature identification and extraction.
3. Complicated pattern matching process, especially for 3–D complex parts.
4. Ambiguities in feature definition, not an unique feature classification.
5. Integration at data/information level not at functional level.

Research into the form of Feature–Based Process Planning systems has been undertaken to solve these problems. Systems such as CIMS/PRO (Iwata et al. 1980), TIPPS (Chang and Wysk 1984), PRICAPP (Pande and Walvekar 1990) and PART (van Houten 1992) use the geometry models, together with user interaction, to identify features and then perform the planning automatically. This approach, being interactive, is not completely automated even though it solves some problems.

Walker and West (1990) describe the use of a Feature Based Methodology to provide a CAD/CAM interface for a manufacturing system producing prismatic machined components. Work concerning machined part representation which relates features to process planning has been sponsored by CAM–I (Computer Aided Manufacturing International) see Bunce et al. (1986), Butterfield et al. (1986), Faux (1986), Pratt and Wilson (1985).

GENPLAN (Gindy et al. 1991) is a feature–based process plan system. In GENPLAN a feature based model describes component geometry and connectivity, and a processing system capabilities model represents the form generating capabilities of machine tools. GENPLAN demonstrates the benefits of these two models for geometric reasoning and set–up determination tasks.

Other examples of CAD/CAM integration through process planning systems are: the XMAPP system that integrates design/process/planning for prismatic parts (Inui et al. 1986), the QTC (Quick Turnaround Cell) system that integrates design/manufacturing/inspection for prismatic parts (Chang et al. 1988), and the AIMSI system that interfaces CAD/CAPP for rotational parts (Wang and Wysk 1988).

2.2.3.3 Features–Based Design Systems

Finger and Dixon (1989) declare that feature–based design systems must be based on a robust modelling system that provides the design engineer with the flexibility to quickly create, modify, and iterate conceptual designs. They must also provide the manufacturing engineer with the precise geometry needed to drive automated equipment and processes. In this approach, the designer is constrained to work with a set of features that have significance for either design, analysis, or manufacturing (CAM–I 1989). The advantage of features is that they maybe able to capture the designer intent. They may encourage standardization leading to improved manufacturability and product quality. They may provide a more convenient design input language, and they will be able to provide designers with manufacturing evaluation and advice that is not available at design time (Luby et al. 1986).

A number of researchers (Pratt and Wilson 1985, Cunningham and Dixon 1988, Shah and Rogers 1988, Cutkosky et al. 1989, Drake and Sela 1989, Devgun and Padmanabhan 1991, Krause et al. 1991, Schulte et al. 1992, Lakko and Mantyla 1993, Case et al. 1993) have developed "Feature–based design systems" to better assist the designer in realizing the product design. Very little work has been done in modelling based on features other than form features modelling. This type of modelling is claimed to help the reasoning process of intelligent systems. Reviews of the status of feature based design systems can be found in Shah et al. (1989), and Case and Gao (1991).

Ishii and Miller (1992) state that feature based design systems have two limitations nowadays: a features library is finite, and not all operations will be possible. No combinations can be created, nor any modifications or deletions performed, which will prevent the various secondary representations from being accurately and uniquely formed. Features design systems seem to solve the data/information integration of future CAD/CAM systems. Nevertheless, features have to be seen only as a possible way to represent design and manufacturing information as an unified entity.

2.2.3.4 Product Model Concept and Computer-Based Environments

The current approaches towards design and manufacturing integration are mainly trying to integrate CAD and CAM systems at the data/information level by transferring product information throughout various separated activities at the design, manufacturing and planning phase. The results are still far from achieving integration at a functional level (Weck et al. 1991).

In order to realize total integration an ideal approach is to integrate all aspects of the information involved in producing a product into a single shared information model. This idea accomplishes functional integration because protocols of communication have to be established between the different activities in order to maintain an "integrated" and "consistent" information model (Bloor et al. 1988). This concept of sharing a product information model among different functional group within an enterprise prompted new directions of research in design and manufacturing integration, namely : Product Modelling (Mantyla 1989). The Product Modelling concept aims to achieve the integration and sharing of data/information among different engineering activities in an enterprise by using a Product Model. In the integration of design and manufacturing information, it is very important to provide a comprehensive framework for dealing with Product Models which can represent, transmit, manipulate and store all the technological information necessary for design and manufacturing activities (Krause et al. 1993). There is not an agreed definition of Product Models, however some definitions are:

"Various kinds of technological information are represented in product models .... basic models are used for describing basic properties and constraints about objects ..... solid models for shape representation, symbolic mathematical formula manipulation for describing various constraints, FEM mesh models ...... engineering models ...... can represent the basic engineering knowledge, such as dimensioning and tolerancing, assembling and kinetic relations, material and manufacturing methods, etc.... and application models are models of products currently designed", (Sata et al. 1985).

"Product modelling is a modelling framework which can capture and represent all the necessary product information through the whole life–cycle of our products, from initial product planning until maintenance", (Kimura and Suzuki 1989).

"Product models can be interpreted as the computer internal logical structured information in a factory, which is available about a product. This informations are integrated into one logical context", (Spur et al. 1989a).

"The idea of an integrated product model bases on the approach to describe all necessary information of the different phases of the product life cycle, their interrelation and product dependant views on the product", (Grabowski et al. 1989).

"PPO schema defines an information model describing the product (form and function), process (activities), and organization (resources).", (Kinstrey et al. 1990).

"software representation of the form and content of the data that describes a product throughout its life–cycle." (Young and Bell 1992).

"The Product Model covers all the information belonging to the part. These are basically geometric information and the representation of the part", (Bjorke and Myklebust 1992).

"... the term product model can then be interpreted as the logical accumulation of all relevant information concerning a given product during the product life cycle", (Krause et al. 1993).

Product modelling, as an essential part of the computer–aided product development activity, yields product model data as its results. The term product development activity refers to those stage or phases from an initial concept to a proven prototype of the product (Krause and Major 1988, Bauert 1989). A product data model, which is generated by product modelling during the product development activity, should be able to support the whole product life cycle concerns (McKay and Bloor 1991). Product Model research has been carried out by academic research institutes to establish product modelling environments and to allow experimentation with the integration of automated applications. Research has been undertaken by:

1. Tokyo University (Sata et al. 1985, Kimura and Suzuki 1986, Kimura et al. 1987, Kimura and Suzuki 1989).
2. Berlin Technical University (Spur et al. 1986, Spur et al. 1989a).
3. University Karlsruhe (Grabowski et al. 1989).
4. Concurrent Engineering Research Center (Kinstrey et al. 1990).
5. Loughborough University of Technology and Leeds University (Shaw et al. 1989, Young and Bell 1992, Corrigall et al. 1992)
6. IMPPACT, ESPRIT No. 2165 (Bjorke and Myklebust 1992).
7. University of Illinois at Urbana–Champaign (Lu 1992).
8. University of Massachusetts at Amherst and University of Karlsruhe (Nnaji et al. 1993).

Krause et al. (1993) present a state of the art review on Product Modelling. In this review it is recognised that product development and product modelling activities are multi–facet subjects determined by many complex factors such as human, organization, product strategy, and available information technology. Therefore the available information and technology is one of the most decisive factors in developing product modelling applications. They also state that the generation of products is dependent on not only the market and customers, but also the available knowledge and information about the capabilities of the factory and suppliers. These authors emphasise that the rapid development of computer aided information technologies has triggered situations where a set of information models (such as factory models that include tool or material flow models, order models, usage models and feedback information) all have to be linked together. Therefore, they conclude that new product modelling approaches have to consider the complexity of product development tasks and their interdependencies. Most of these product model computer–based environments have evolved into the support of new engineering practices such as Simultaneous Engineering and Life Cycle Design of Products. In the next section a review of these two contemporary practices is presented.

2.3. Contemporary Engineering Practices

Simultaneous Engineering focuses on integration of multi–disciplinary expertise, cooperation among competitive perspectives, communication of product life–cycle concerns, and coordination of group problem–solving activities. The product life cycle design concept extends the development considerations from technical aspects to the limited resources of our world, focusing on environmental protection and occupational health issues. A review of these contemporary practices and their information system support is presented in this section.

2.3.1. Simultaneous Engineering

The philosophy of Simultaneous Engineering (SE) or Concurrent Engineering (CE) has been proposed as a potential means of improving product development practice (Sohlenius 1992). This philosophy involves simultaneously satisfying the functionality, reliability, produceability, and marketability aspects of new products in order to reduce product development time and cost, and to achieve higher product quality and value. Nevins and Whitney (1989) argue that Simultaneous Engineering seems to be the key to achieving and sustaining a competitive advantage through the synergy of integrated product and process design whilst also considering multiple life cycle factors, such as functionality, serviceability, manufacturability, marketability and recyclability.

In this philosophy the objectives focus on (Winner et al. 1988, Painter et al. 1991):

- improvement of quality
- reduction of life–cycle costs
- reduction of development lead times

Simultaneous Engineering has been defined by Winner et al. (1988) as "a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements".

Cleetus (1992) has proposed a new definition: "CE is a systematic approach to integrated product development, that emphasizes response to customer expectations and embodies team values of cooperation, trust, and sharing in such a manner that decision making proceeds with large intervals of parallel working by all life–cycle perspectives, synchronized by comparatively brief exchanges to produce consensus".

In both definitions the importance of addressing the different aspects of SE in a systematic manner is emphasized. In order to be systematic, Simultaneous Engineering should be characterized by applying principles which introduce cultural, human and organizational changes within the enterprise through the use of formal methodologies, in some cases, supported by information technology. Painter et al. (1991) defines the SE principles which guide the enterprise changes have been classified as follows:

- Organization principles: new organizational structures, customer focus attitude, discipline, continued commitment, leadership and teaming.
- Process Improvement principles: integrated and concurrent development of activities for product life cycle, and continuous process improvement.

- Information Management principles: higher levels of information and knowledge integration, enhancement of information and knowledge communication, and management of corporate resources (people's knowledge, information technology, etc.).

Linton et al. (1992) state that the application of the above principles will help to ensure that all aspects are considered during the implementation and operation of SE. The extent to which each of the principles is adopted does however tend to vary depending upon the nature of the organisation (Hon 1992). The author has identified two kinds of organisations which are committed to SE related research, namely industrial enterprises and academic research laboratories.

Industry is often forced to adopt a pragmatic approach to SE in order to solve their more immediate problems. Hartley (1990) recognised that organisational and process improvement principles are usually tackled first as they provide highly visible benefits for low–capital investment. This approach can also be justified because of the lack of any complete commercial computer support solutions and the possible trepidation associated with software solutions which is often due to negative past experiences (Craig 1991, Byrd et al. 1992).

Academic institutions and research laboratories are commonly divided between providing organisational and technological support for major change initiatives. The former research has concentrated on developing methodologies for the introduction of both new organisational structures and team work (Barret 1994). The latter has been closely related with developing technological solutions to support SE. This work has focused largely on developing software applications to support the implementation of specific process improvement techniques and in developing frameworks that allow the capture and sharing of cross–functional information (Kahaner and Lu 1993). The goals of such information technology research are long–term and are intended to enhance the understanding of future SE support techniques and to smooth the transgression from current information technology solutions to those of the future. This review focuses on these latter academic initiatives, especially the approaches related to the development of computer aided support tools. Nevertheless, the subsection below is dedicated to presenting a general overview of enterprise initiatives in order to outline the principal issues.

2.3.1.1 Enterprise Initiatives for Simultaneous Engineering

Leading manufacturing companies are becoming aware of the immense benefits which can be derived from SE. The main barrier that has to be overcome in applying SE is the need to breakdown organisational and cultural barriers (Tucker and Leonard 1994). Currently, the introduction of SE in industry usually has the following phases (Bishop 1991, Voss et al. 1991, Craig 1991):

1. Review of SE philosophy and strategy for product development
2. Organization of responsibilities and the coupling of activities based on team working principles.
3. Introduction of process improvement principles using formal methods and techniques rather than using advanced information technology solutions.
4. Adapting existing engineering applications to better support concurrent product development
5. If considered necessary, acquisition of information technology.

The use of advanced information technology solutions seems to play a minor role in the introduction of SE within industry. Organizational issues take priority (Dunn 1990, Siegal 1991, Woodgate 1991, Wheeler et al. 1991), followed by the use of formal methodologies, such as design for assembly and manufacture (Stoll 1988, Hiatt 1990, Miles 1990, Nichols 1991, Gerhardt et al. 1991, Lee–Mortimer 1991, Booty 1991, Eversheim and Gross 1991, Corbett et al. 1991), and a range of quality engineering techniques for managing complex system trade–offs and for finding optimum design and production process parameters (Henshall 1989, Denton 1990, Potts 1990).

However, this is the current situation and it is likely that as companies become more experienced in SE, they will start to look for more sophisticated information technology tools. In a survey undertaken by Stevenson (1992) current information technology tools required for Simultaneous Engineering were identified and categorized into four areas:

1. Product Information Management (PIM): PIM tools are configuration management information systems which organize data produced on workstations to serve engineering, analysis, manufacturing, and management.

2. CAD Frameworks: these are specific application frameworks to improve performance during the design process and are based on a set of standards for CAD data exchange such as IGES and STEP (ISO CD 10303 – 1).

3. CSCW (Computer Support Co-operative Work): a CSCW tool is an information system which supports team work by providing enabling technologies such as computer networking and it's associated hardware, software, services and techniques.

4. Electronic Design Engineers Notebooks: these are hypermedia systems that readily permit the extraction of information from past engineering records.

The results of this survey showed that only the first two categories are supported by commercial products, such as Production Information Management systems and CAD Frameworks (Johnson 1987, Lawrence 1990, So et al. 1992). In the latter two categories, the user need exists but software vendors have yet to produce commercial products. Prototype systems developed in research projects to explore these ideas do however exist (Wilson 1990, Lu 1992, Reddy et al. 1993, Cutkosky et al. 1993).

2.3.1.2 Academic Initiatives for Simultaneous Engineering

A broad range of research related to SE has been undertaken by academic institutes. Kahaner and Lu (1993) has defined the following key technological requirements of Simultaneous Engineering:

1. Modelling Methodologies

2. Computer Aided Decision Support

3. Information System Architectures

A diversity of methodologies is needed to model the important aspects of the product realization activities such as information, processes, organization, etc. Research effort dedicated to defining methodologies for organisational and cultural issues of SE introduction can be found in Evans (1990), Gillen and Fitzgerald (1991), Karandikar et al. (1992) and Evans et al. (1994). The complexity of co–ordinating and structuring the design and development of support systems for SE necessitates the use of formal modelling methodologies for their representation. These methodologies include techniques to model data, information and processes. Modelling information has been considered a key in the development of integrated manufacturing information systems (Chadha et al. 1991). Section 2.4 is dedicated to review in detail the modelling methodologies which have been developed to facilitate the design and implementation of information systems for complex manufacturing environments. The idea of combining modelling methodologies in integrated frameworks for the development of system concepts has resulted in the creation of refence models for Enterprise Integration. Section 2.5 describes the frameworks which have been developed to tackle the Enterprise Integration issue.

Computer Aided Decision Support applications assist specific engineering tasks for example: plastic part design (Gadh et al. 1989, Ishii 1992, Hambaba et al. 1992), process planning (Cutkosky et al. 1989, Lu 1992), printed wiring board design (Bowen et al. 1993, O'Grady et al. 1991), mechanical design (Rehg et al. 1988, Finger et al. 1992, Wu et al. 1992), etc. These computer systems have been developed as isolated applications (Rehg et al. 1988, Gadh et al. 1989, Vujosevik and Kusiak 1991, Ishii 1992, Bowen et al. 1993, O'Grady et al. 1991, Hambaba et al. 1992), or as part of integrated environments for SE (Cutkosky et al. 1989, Lu 1992, Wu et al. 1992). The development of the latter environments is usually based on an Information System Architecture. These systems are open and distributed computer–based architectures which provide different integration services and allow the communication and exchange of information among the decision support applications (Blinn et al. 1991) The emphasis in this work is on defining the integration frameworks and the communication protocols between the system elements for the support of designers in a Simultaneous Engineering Environment.

There has been considerable research and development activity to establish standard Information System Architectures for large–scale engineering and manufacturing businesses. Only work undertaken into developing Information System Architectures related specifically to simultaneous engineering is relevant to this review (e.g. Genesereth et al. 1992, Lu 1992, Reddy et al. 1993, Cutkosky et al. 1993). Other endeavours to develop such architectures include IISS (Judson 1986), IDS (1989), CFI (Painter 1990), EIS (1990) and CIM–BIOSYS (Leech et al. 1991). However, these architectures are not specifically designed for the support of simultaneous engineering and will therefore not be considered further in this review.

2.3.1.3 Computer Aided Simultaneous Engineering Systems

A detailed review of Computer Aided Simultaneous Engineering Systems can be found in Molina et al. (1994), see appendix A. This review paper considers a wide range of published research work on enabling technologies.

2.3.2. Product Life Cycle Design

Product life cycle design is becoming the backbone of a new industrial culture named sustainable development. Sustainable means that products should be designed for their whole life cycle i.e. production, distribution, usage and disposal with minimized influence on the environment, occupational health and use of resources (Alting and Jorgensen 1993). In other words, this concept extends the development considerations from technical aspects, as conceived by Simultaneous Engineering, to the limited resources of our world, focusing on environmental and human protection.

2.3.2.1 Life Cycle Engineering Philosophy

Colby (1990) recognises that there is a growing awareness of issues such as environmental protection, human centred technologies and resource utilization, which will in future undoubtedly play a much larger role in the customers evaluation of a specific product. Tipnis (1993) states that to accommodate these new demands, and remain world class, a new approach to product realization has been introduced. Alting (1991) introduces the Life Cycle Design of Products concept which aims to achieve a competitive advantage through the sustainable development of high quality products. Als Pedersen and Alting (1991) claim that Life Cycle Design of Products, together with Simultaneous Engineering, complete the total concept of Life Cycle Engineering which promises to offer an industrially applicable solution through the adoption of a new and all encompassing philosophy. The life cycle approach implies that products are planned and developed for all life cycle phases (production, distribution, use and disposal) before production and marketing. Alting (1991) has identified the most important criteria which have to be considered when performing Life Cycle Engineering as:

- Product Properties: the properties of the product are numerous and may include function, quality, reliability, cost, etc. These properties reflect how well the product fulfils the established technical specifications and the associated cost limitations. The specifications must reflect the actual wants of the customer. Some techniques applied to help ensure customer satisfaction are ISO 9000 for Quality Management Functions, and paradigms for establishing environmental specifications discussed in Remmerswaal (1990).

- Ease of Manufacture: The manufacturability of a product can be measured against a number of criteria e.g. standardisation; ease of production, assembly and automation; flexibility and reliability. Some of the techniques used to improve the manufacturability of evolving designs are Design for Manufacture (Booty 1991) and Design for Assembly (Boothroyd 1985, Boothroyd and Alting 1992).

- Resource Management: Nowadays there is a need to optimise the utilization of resources (material and energy) and processes. Due to dwindling natural resources the use of unsustainable resources in products is becoming unacceptable. The minimisation of material and energy consumption not only reduces costs but also appeals to the general consumer. The improvement of manufacturing processes can contribute to a better utilization of resources. Some techniques which support process improvements are: business process re–engineering (Graefe and Chan 1993), enterprise modelling (Williams et al. 1993), material and energy flow analysis (Zust and Wagner 1992), and manufacturing process analysis (Byrne and Scholta 1993).

- Environmental Protection: Environmental considerations ensure that a product can be produced, distributed, used and disposed/recycled without unnecessary harm to the environment. The development of methodologies/tools that enable the assessment of the impact of a certain product on the environment are needed. The most common techniques are: Design for Environmentally Friendly Production (Tipnis 1993) and Design for Recycling and Disassembly (Jovane et al. 1993).

- Human Centred Systems: Technology is not the only important factor in Life Cycle Engineering. Organisation and people issues influence the use of technologies. These issues have to be researched to design, plan and operate new human centred production systems in order to create working environments that consider occupational health factors (Kidd 1992).

Alting and Jorgensen (1993) have defined sustainability as a key current requirement of the manufacturing community. Several researchers have been engaged on this area of research, for example: Alting (1991), Ebach et al. (1992), Zust and Wagner (1992), Boothroyd and Alting (1992), and Jovane et al. (1993). Different enabling technologies have been identified to support the life cycle concept. The next subsection is dedicated to review such information technologies.

2.3.2.2 Enabling Technologies in Life Cycle Engineering

Enabling technologies have been developed to support the implementation of Life Cycle Engineering Techniques. The Life Cycle Center at the Technical University of Denmark (Alting and Jorgensen 1993) has identified the following information technologies to support Sustainable Industrial Production :

- Tools that support decisions made throughout the product life–cycle by providing multi–criteria evaluations, simulation, or feedback, with databases and integrated CAD/CAM systems for support of the design process including use of environmental/resource guidelines.
- Integrated Information Systems to support the extended functions of sustainability, inlcuding decentrilized, distributed production and disposal, logistic covering all the life cycle. These system should be able to capture complete life–cycle information needs by using adequate product structures.
- "Green" CIM Systems to allow the tracking of products, management of service/maintenance activities and transportation of products for disposal, documentation of products/material, etc.

Alting and Jorgensen (1993) argue that the combination of Life Cycle Design of Products and Simultaneous Engineering seems to accomplish a holistic approach for product development. Boothroyd and Alting (1992) define the need for new information technology applications which ensure information integration throughout the life cycle. The complexity of such information support systems necessitates the use of formal methods for their representation i.e reference models, modelling techniques and advanced computer tools. In the next section 2.4 the modelling methodologies used to develop information integrated systems in manufacturing are reviewed.

2.4. Relevant Modelling Methodologies

A modelling methodology refers to a class of similar methods, where a method is an organized, single purpose discipline or practice (Coleman 1989). Different methods have been developed to assist in the modelling of different aspects of information systems which support manufacturing environments. This section gives a brief description of modelling methodologies. They have been classified in four groups:

1. Data/Information Modelling Methods: Data Flow Diagrams (DeMarco 1979, Yourdon and Constantine 1979, Orr and Gane 1989), Entity Relationships Diagrams (Chen 1976, 1979), IDEF1 and IDEF1x (Appleton 1985, Bravoco and Yadav 1985a, 1985b), NIAM (Verheijen and VanBekkum 1982, Bray 1988), Dependency Diagrams (Smith 1985), and EXPRESS (ISO DIS 10303–11).

2. Processes Modelling Methods: IDEF0 (Bravoco and Yadav 1985a, 1985c), SAMM (IPAD 1977a, IPAD 1977b) and IDEF3 (Mayer et al. 1990).

3. Behaviour Modelling Methods: IDEF2 (Bravoco and Yadav 1985a, 1985d) and Petri Nets (Murata 1989).

4. Hybrid Modelling Methods: Semantic Nets (Amble 1987), Conceptual Graphs (Sowa 1984), and Object–Oriented methods (Wirfs–Brock et al. 1990, Coad and Yourdon 1990, Booch 1991, Rumbaugh et al. 1991).

The review does not include all the modelling methodologies, however the more popular methods used in the development of information systems which support engineering activities are presented. Detailed description of each of them can be found in the literature referenced.

2.4.1. Data/Information Modelling Methods

Data modelling allows the description of the information structure relevant for a system in an implementation–independent format named a data model (Scheer 1989). Data modelling methods have generally derived as aids to database design. As such, they tend to support the modeling of entities, and the relationships between entities. An entity is defined as being something which exits and is distinguishable, a set of attributes is generally attached to the entity. Relationships may also possess attributes which properly belong to the relationship itself, rather than to any of the associated entities (Hars and Scheer 1991). Information modelling is an outgrowth of data modeling (Schenck and Wilson 1994). The goal of information modelling is to characterize real world objects as completely and realistically as possible. Information modelling is related with the identification, representation and composition of the data, information and knowledge that describes a real object or objects. The results of this modelling activity is an information model (Klein and Scheer 1990). There are basically two differences between data and information modelling. Data modelling is targeted to produce a data model that is computer processable, the information model is not, but could be computer processed. Second, an information model must be made explicitly and formally documented, in the data model rules are implicit and informally documented. Therefore data modelling techniques can be used to develop information models if the appropriate formal documentation is generated during the modelling exercise (Eastman 1991).

Data Flow Diagrams

Data Flow Diagrams (DeMarco 1979, Yourdon and Constantine 1979) provide a view of the data flows in a system and the various data stores and processes/functions to/from which they flow. This is a popular method in information system design and appears often in Computer–Aided Software Engineering tools (Orr and Gane 1989). It is generally used to determine the information contained within a system. It can be decomposed hierarchically as in IDEF0, with the attendant benefits, but does not clearly show the process flow in a manufacturing system. It does not incorporate any constraints in the data and data relationships.

Entity Relationships

Entity Relationships (ER) diagrams (Chen 1979) provides a view of data entities and their associated relationships. An important characteristic of this method is its simplicity. The information entities can be represented in a real world manner. The mapping of ER diagrams into a relational database design is simple and relatively straightforward (Teory et al. 1986). Extended ER allows the definition of data abstractions but neither ER nor

Extended ER has provisions for incorporating constraint and exceptions (Smith and Smith 1977).

IDEF1/IDEF1x

The IDEF1 and IDEF1x (Appleton 1985, Bravoco and Yadav 1985a, 1985b) methods are similar to ER conceptually, but their graphical representation is different. IDEF1 and IDEF1x are more complicated semantically, thus making them more difficult to use than entity–relationships. Nevertheless the IDEF methodology is more integrated within itself (IDEF0, IDEF1x, and IDEF2) than other methodologies.

NIAM

The NIAM (Nijssen's Information Analysis Modelling) methodology (Bray 1988) is another data modelling method. This method allows to model constrains , in addition to showing objects and their relationships (Verheijen and VanBekkum 1982). NIAM is simple to learn and use, and can be mapped into a relational database design. NIAM is one of the most popular data modelling methods used in engineering (Bjorke and Myklebust 1992)

Dependency Diagrams

The Dependency Diagram (Smith 1985) is a combination of a dependency list and diagram, in which the dependencies of data elements are described. The use of this modelling methods is a fairly rigorous exercise, especially the construction of the dependency list. The procedure involves the determination of information requirements and then the listing of the dependencies. Finally, the diagram is generated from the list, and simplified. The diagram can be directly mapped into a relational database design. There is no vehicle for constraint and exception incorporation.

EXPRESS

The STEP (STandard for the Exchange of Produc model data) has defined a data definition language know as EXPRESS (ISO DIS 10303–11). EXPRESS is not a modelling techniques but it is a language that allows the development of information model specification. EXPRESS supports the separation of abstract and concrete ideas, since different users have need for different degrees of detail. EXPRESS also models the constraints which are to be imposed upon the things which are modelled and the operations in which the things modelled will participate (Schenck and Wilson 1994).

2.4.2. Processes Modelling Methods

The description of system's functions through the process of function decomposition and categorisation of the relations between functions is provided by Process Modelling (Klein and Scheer 1990). A process model depicts how a certain activity is performed in multi-staged functional levels and what constraint are associated to them. Process modelling represents how an entity does something in terms of strategies, rules and constraints (Busby and Williams 1993).

IDEF0

The IDEF0 methodology (Bravoco and Yadav 1985a, 1985c) is a top–down hierarchical method which provides a description of functions and processes in manufacturing. It does have a capacity to incorporate constraints and mechanisms. It has similarities to flow–charting in that there is a series of processes/functions arranged sequentially. The hierarchical breakdowns allows for defining the system in any number of levels of detail, down to the level required for analysis. This makes it easier to understand complex manufacturing systems (Sarkis and Lin 1994). A state of the art review of IDEF0 can be found in Colquhoun et al. (1993).

Systematic Activity Modeling Method (SAMM)

The SAMM modeling methodology is a top–down hierarchical method, similar to IDEF0 (IPAD 1977a). It uses a system of nodes and branches, each node being represented by an activity diagram. Activity and data flows together with the amount of data being exchanged is represented by this method. The data flows describe the relations between the activity diagrams. The IPAD studies used this method to model the design process (IPAD 1977b).

IDEF3

The IDEF3 methodology (Mayer et al. 1990) is a relatively recent development. It was designed to capture the knowledge of an expert about how a particular process, event, or system works. It is a language which allows to describe processes. A representation of what a system is doing can be captured by using IDEF3. It incorporates constraints in the processes, and it does have a hierarchical decomposition similar to IDEF0.

2.4.3. Behaviour Modelling Methods

These modelling techniques are concerned with the dynamic behaviour of systems (Kinstrey et al. 1990). Behaviour modelling describes the dynamics of a system. i.e. what is happening through time. All entities in a system have some state in time, execute an operation, and exhibit some performance. Modelling the behaviour allows the representation of time dependencies (sequence, parallelism, concurrency), operation execution and performance (Czernik and Quint 1992). Modelling methods are under development, however there is currently no methodology which adequately describes dynamic behaviour (Harhalakis et al. 1992).

IDEF2

The IDEF2 methodology (Bravoco and Yadav 1985a, 1985d) provides a description of the dynamic aspects of a system: the resources used to produce a product (e.g. manufacturing facilities), the paths an entity can take and the resources needed along the path (e.g. material flow), status of resources (machine use), depending on system status, and controls on activities (Banerjee and Al–Maliki 1992). IDEF2 ended up being a simulation modelling tool (Mayer et al. 1990).

Petri Nets

Petri nets are a graphical and mathematical modelling tool applicable to many systems. Petri nets allow the description and study of information processing systems which are characterized as being concurrent, asynchronous, distributed, parallel, non–deterministic, and/or stochastic (Murata 1989). Petri nets, as a graphical tool, is similar to flow charts, block diagrams, and networks which can be used as aids for visual communication. In addition, tokens are used in these nets to simulate the dynamic and concurrent activities of systems. As Petri nets are a mathematical tool, it is possible to set up state equations, algebraic equations, and other mathematical models governing the behaviour of systems (Peterson 1981). Detail applications of Petri Nets to Manufacturing Systems can be found in DiCesare et al. (1993).

2.4.4. Hybrid Modelling Methods

The hybrid modelling methods allow a system to be described by modelling data, function and behaviour in combination. The objective of these modelling methods is to characterize real world objects as completely and realistically as possible. The integration of different modelling dimensions in hybrid methods enable the construction of models which are a reflection of reality.

Semantic Nets

A semantic net is a formalism for representing facts and relations between facts with binary relations (Amble 1987). Semantic nets are extensively used in artificial intelligence applications. Each object modelled is represented as a node in the network and these networks can become quite large. These can model data abstraction as IS–A and PART–OF relationships. These semantic nets are the basis for Conceptual Graphs (Sowa 1984).

Conceptual Graphs

A Conceptual Graph is a knowledge representation notation or language (Sowa 1984). Conceptual graphs are derived from canonical formation rules which are a context–free graph grammar. Conceptual graphs can be mapped into formulas in the first–order predicate calculus. A Conceptual Graph is a combination of concept nodes and relation nodes where every arc of every conceptual relation is linked to a concept. Entities, attributes, states, and events are represented by concept nodes, and the interconnections between them are relation nodes.

Object–Oriented Methodologies (OOM)

The Object–Oriented modelling methodologies (Wirfs–Brock et al. 1990, Coad and Yourdon 1990, Booch 1991, Rumbaugh et al. 1991) introduce a new notation, rather than entities and relationships, an object modelling notation models objects and relationships, where an object is both an entity and the operations which can be performed on that entity. It allows for data abstraction, inheritance, information hiding, dynamic binding and polymorphism. The interfaces between the objects need to be well–defined because that is the only way data within an object can be accessed. An object is a variable. In classical object–oriented programming (Stroustrup 1986), this variable has a type (called a class) defining its structure (data/information) and operations (methods) which may be performed on the object. Object oriented methods have become very popular in the implementation of computer based applications that support manufacturing systems (Adiga 1993, Nof 1994).

Although all these methods have been proposed to guide the development of information systems. These methodologies do not satisfy all the requirements of the manufacturing system environments. They lack a comprehensive formal framework to support the complete life cycle of an integrated system i.e from concept through implementation. Therefore reference models or architectures for integrating manufacturing activities and enterprises have been created for the task of developing information integrated systems. The following section reviews the most relevant reference models found in the literature.

2.5. Reference Models for Integrating Enterprises

Different reference models, frameworks and architectures have been developed to be used in information and manufacturing system development. Within the manufacturing field, reference models have been defined to classify, evaluate and develop CAD systems (Finkenwirth and Jansen 1989), to model and implement CIM systems (BSI PD 6526: 1990), and to assist in the development of future standards (BSI DD 203: Part 1 : 1991, DIN Technical Report No. 15).

The IFAC/IFIP Task Force report on Architectures for Integrating Manufacturing Activities and Enterprises (Williams et al. 1993) found suitable for the task of describing an integrated system, its life cycle and the methodology for its application the following reference models: CIMOSA (ESPRIT Project 688/5288), GRAI–GIM (Doumeingts et al. 1992) and Purdue Enterprise Reference Architecture (Williams 1991). This Task Force is aiming to develop a Generic Enterprise Reference Architecture and Methodology (GERAM) on the basis of the previously analysed architectures (Bernus and Nemes 1994). In addition to these architectures, other endeavours have developed their own frameworks, for example: Integrated Enterprise Modelling (Mertins et al. 1992), TOVE (Fox 1992), Enterprise Modelling System (Graefe and Chan 1993) and the Stair–like Reference Architecture (Chen et al. 1994). Work in the area of developing standards for Open Distributed Systems has originated the Reference Model for Open Distributed Processing or RM–ODP (ISO/IEC 10746–1). However, in this literature survey only the CIMOSA, GRAI–GIM, Purdue Enterprise Reference Architecture (PERA) and the Reference Model for Open Distributed Processing (RM–ODP) are considered as they represent the more complete and well documented architectures.

2.5.1. The concept of reference models, frameworks and architectures

The terms framework and architecture have been used ambiguously within the manufacturing domain to denote reference models that assist in the development of integrated systems during different phases of, or throughout, the complete life cycle. Mayer and Painter (1991) pointed out the difference between a framework and an architecture. The latter only denotes the information system architecture in terms of databases, networks, operating systems and integration utilities required for the enterprise integration. The framework refers to an organized representation of characterized situation types that occur during an information system life cycle for enterprise integration. Each situation specifies tasks, methods and tools which can be used to support a particular development situation (Zachman 1987).

Based on these characteristics, different so called architectures for enterprise integration (e.g. CIMOSA, GRAI–GIM and PERA) can be considered as frameworks for enterprise integration. This is because those architectures are used as a reference for the definition of situations and methods that can exist, and be used, in the development of integrated enterprise systems (Williams et al. 1993). In this review, to avoid any misunderstanding, all of them are referred to as reference models for enterprise integration. It can therefore be deduced that a reference model for enterprise integration allows the description of an information integrated system and the modelling of its life cycle.

Reference models provide general representations of different aspects of a system. These representations can be referenced to assist in the development of a system during various stages of its life cycle (e.g. requirements elicitation, system modelling, design and implementation).

There are some characteristics which are common to these reference models (BSI DD 203: Part 1 : 1991). A reference model must be:

1. Structured: based upon readily available and acceptable terminology, methodologies or standards.

2. Flexible: able to be applied to wide range of systems within its domain of applicability.

3. Generic: independent of any existing implementation.

4. Modular: open–ended in its ability to be extended in order to incorporate new concepts and technologies.

The terminology, methodologies or standards used in a reference model determine its structure and content. Reference Models have to use a widely accepted, proven terminology and syntax in order to allow easy identification and interpretation of their structure and content e.g. Views, Levels, Architecture (Menzel et al. 1992). The modelling methodologies provide the notation and syntax to describe the system characteristics e.g. data models based on an Entity Relationship Model (Scheer 1989), processes models based on IDEF0 (Bjorke and Myklebust 1992). The methodologies used in the reference models should enable them to be translated into different notations and to be modified. The use of standards within a reference model is likely to increase its uptake and ensure the usefulness and compatibility of the individual representations within the model. Nevertheless reference models are not 'de–facto standards', but they do provide a more structured and integrated guide for the development of systems (Hars et al. 1992).

The characteristic of flexibility enables the development of a range of systems, within a given domain, by customizing the reference model according to the needs of the user (Goranson 1992). Usually reference models are targeted at a particular type of system or at a specific element of the systems description. They can therefore be classified as either system–oriented models (e.g. CAD, CIM, etc.) or focus–oriented models (e.g. data, process, function, etc.).

The reference models are generic enough to support the description of a wide range of systems independent of any configuration or existing technologies (Bernus et al. 1994). They do not focus on a particular system but on the structures which are typical to a set of systems (e.g. CAD, CIM, etc.) and their common attributes i.e. information system architecture, data exchange formats, organization structure or configuration (Finkenwirth and Jansen 1989, Lienhart 1990).

The modular characteristic of a model is necessary if it is to be extensible and allow the incorporation of new concepts and technologies (Spur et al. 1989b). Flexibility is achieved by defining different views or viewpoints in the reference model that correspond to the aspects of the system that is being described e.g. enterprise, information, computation, etc. (Linington 1992). In this way, new technologies can be adopted by identifying their individual aspects and mapping them into the views defined by the reference model. The viewpoints can be considered as sub–reference models defined within a reference model in order to describe more specialized aspects of a system e.g. an information viewpoint which contains reference models related to reference data, functions or processes (Klein and Scheer 1990).

CIMOSA, GRAI–GIM and PERA are based on the reference model concept outlined above. The different terminologies used by these reference models to address similar concepts introduce a kind of chaos when an analysis of their characteristics has to be made. Among the most important characteristics required in any reference model are (Bernus et al. 1994):

1. Formality in its approach to model the life cycle of a system
2. Existence of an underlying methodology for its application
3. Definition of suitable methods and tools to model different views
4. Specification of an Integration Platform or Integration Infrastructure

The characteristics defined above represent the frame where the above reference models can be mapped to be analysed.

2.5.1.1 Life Cycle System Modelling

The reference models for enterprise integration allow the modelling of the system evolution during its life cycle (Williams et al. 1993). These reference models show the structure and relationship of all of the tasks involved in the concept, analysis, design, building, commissioning and operation of the desired integrated enterprise. If the system is too complex, a representation of the system at different phases of the life cycle can be produced. Each representation has to be characterized by a model in terms of the important attributes of the system at that particular phase (ESPRIT Consortium AMICE 1989). The consistency of these models and results for each life cycle phase must be constantly checked. The following life cycle phases have been defined: conceptual definition, requirements, design, implementation, operation and maintenance (Williams 1991).

2.5.1.2 Underlying Methodology

The underlying methodology in the context of a reference model for enterprise integration is a detailed process model, with guide-lines of how to perform the development activity. The methodology should cover the entire reference model application and should lead the developer during the integration process (Bernus and Nemes 1994).

2.5.1.3 Methods and Tools

Different methods are defined within the reference models to assist in the modelling of multiple views. These multiple views represent important aspects about a system which have to be modelled (ICEIMT Workshop IV 1992). Many different aspects of a system can be characterised in an model, for example: information, function, organization, resources (ESPRIT Project 688/5288). The modelling views offered should cover a minimal set (e.g. CIMOSA views) but this set should be expandable with new related views. Modelling views should be based on a common theory, or meta–model, through which views can be related (Jorysz and Vernadat 1990a, 1990b, Vernadat 1992). The various modelling methods used to describe the views and the corresponding classes of models have been reviewed in section 2.4. Modelling tools are software systems designed to support the application of a method, for example: EXPRESS compiler and graphical editor (STEP Tool Kit 1993), IDEF software (META Software Corporation 1990), and RationalROSE (Rational ROSE 1993) software which supports the Booch object oriented methodology.

2.5.1.4 Integration Platforms or Integration Infrastructures

Integration platforms or integrated infrastructure represent the conceptual and physical structures of some of the components in an enterprise, such as computers, networks, manufacturing facilities, etc. (Mayer and Painter 1991). These type of architectures are an integral part of the reference models for enterprise integration because they denote system configurations (e.g. information system or manufacturing system architectures) capable of fulfilling the functional requirements. Examples of such architectures include IISS (Judson 1986), IDS (1989), CFI (Painter 1990), EIS (1990), CIM–BIOSYS (Weston 1993), IIS (Querenet 1992), and the information system architectures for Simultaneous Engineering reviewed in appendix A.

2.5.2. CIMOSA

The Open System Architecture for Computer Integrated Manufacturing (CIMOSA) was developed by the European CIM Architecture Consortium (AMICE) under the ESPRIT projects 688, 2422 and 5288. The ESPRIT project AMICE was carried out by 22 European CIM users, vendors, implementor and research organisations (Kosanke 1992). This project defines and develops an architecture (CIMOSA cube) for the definition, specification and implementation of Computer Integrated Systems. Project results are summarised in "CIMOSA : Open System Architecture for CIM" by Springer (ESPRIT Project 688/5288) and several publications by project members (Jorysz and Vernadat 1990a, 1990b, Klittich 1990, Kosanke 1992, Vernadat 1992, Querent 1992). The CIMOSA architecture is the base of the European Pre–Norm CEN/CENELEC ENV 40 003 (BSI DD 194:1990) on the Modelling Framework which also has become the base for the international standardisation work in ISO TC184/TC5/WG1.

CIMOSA consists of a Modelling Framework supporting the representation of enterprise operation requirements, design and implementation. Therefore the CIMOSA life cycle includes requirements, design, implementation, release, operation and maintenance (ESPRIT Project 688/5288 pages 16–17). According to the defined phases the models which can be represented are: requirements, design and implementation models. In CIMOSA, the initial phase regarding the conceptual definition, where the business and manufacturing strategies are considered, is omitted. Nevertheless it is assumed that this information is available. CIMOSA provides a mapping between the remaining life cycle phases and models generated, where the implementation model addresses the last three phases (Kosanke 1992).

The modelling framework shown in figure 2.1 consists of a reference architecture and a particular architecture which describes the structure of a particular enterprise (Jorysz and Vernadat 1990a). The contents of the reference architecture are used to engineer models of enterprise operations. The reference architecture (generic and partial level) provides the set of constructs which enable the operation of a particular enterprise to be modelled. Generic constructs (Generic Building Blocks and Building Block Types) are applicable to all industrial enterprises whereas more specific macro constructs (Partial Models) are aimed at modelling specific enterprise domains and special industries (ESPRIT Project 688/5288 pages 49–51). These building blocks and the macros are used to create and maintain the enterprise models. These concepts conform to the meta–model of CIMOSA which supports the underlying methodology for the application of the CIMOSA architecture (Vernadat 1992).

CIMOSA defines four different views to comply with the common practice of focusing on different aspects of the model rather than looking at the model as a whole. CIMOSA provides a set of constructs to represent the Function, Information, Resource and Organization views for the different modelling levels i.e. requirements definition, design specification, and implementation description model (Jorysz and Vernadat 1990a, Jorysz and Vernadat 1990b, ESPRIT Project 688/5288 pages 51–72).

In order to increase the potential for executability CIM–OSA has defined two major environments, the Enterprise Engineering Environment and the Enterprise Operation Environment, and a set of specified system services know as the Integrating Infrastructure – IIS (Klittich 1990). The first of these environments formalizes the development of enterprise models and their conversion to working programs for the system. The second formalizes the testing, proving and acceptance of the resulting programs as new additions or changes to the operating systems (Querent 1992). The IIS defines how all such programs, as just noted, work together to carry out the overall functions of the integrated computer system (ESPRIT Project 688/5288 pages 83–115). Mapping IIS to existing standards (ODP, OSI) is currently done to identify requirements for alignment with, and adaptation of, currently available or emerging standards (Kosanke 1992).

2.5.3. Purdue Enterprise Reference Architecture

The Purdue Enterprise Reference Architecture and the related Purdue Methodology was developed at Purdue University as part of the work of the Industry – Purdue University Consortium for CIM. This latter work stated formally in 1989 but bears on the Purdue Reference Model developed starting in 1986 and earlier work of the Purdue Laboratory for Applied Industrial Control dating back to the mid seventies (Williams et al. 1993).

Figure 2.2 presents a simple block diagram form of Purdue Enterprise Reference Architecture (PERA). The life cycle of PERA includes: concept, definition, design, construction and installation, and operation phases. PERA introduces a set of layers related to each of the life cycle phases (i.e. concept, definition, specification, detailed design, manifestation, and operation layers). Two views are defined to differentiate functional specifications (i.e. concept and definition phases) from implementation aspects (i.e. design, construction and installation, and operation phases). The functional view corresponds to the analysis of requirements where all tasks and functions are defined regardless of how they are implemented. The implementation view embraces the remaining phases in terms of the human organizations, and the physical hardware and software to be used (Williams 1991).

The conceptualization phase (concept layer) begins with the CIM Business Entity which leads first to a description of the management's mission, vision and values for the entity plus any further philosophies of operation or mandated actions concerning it such as choice of processes, vendor selection, etc. In the manufacturing plant the above prescription and selection by management of possible options leads to the establishment of operational requirements for the plant. This latter then leads to a statement of requirements for all the equipment and for all of the methods of operation, etc., for these units. These are developed in the Definition Layer (Williams 1993) Note that there are two, and only two kinds of requirements developed from the management pronouncements – those defining information–type tasks and those defining physical tasks. Tasks become collected into modules or functions and these in turn can be connected into networks of information or of material and energy flow. These latter then form the Information Functional Network or the Manufacturing Functional Network respectively as shown in figure 2.2 Note also that no consideration of implementation methods or of the place of humans in the system has yet taken place.

Once implementation is considered, the first need is to define which tasks, on either side of the overall architecture, will be fulfilled by people. By so doing, PERA defines the place of the human in the Information Architecture and also in the Manufacturing Architecture. These together form a Human and Organizational Architecture. The remainder of the tasks of the Information Architecture then define the Information system Architecture (all the tasks performed by the computers, software, databases, etc). The remainder of the tasks of the Manufacturing Architecture define the Manufacturing Equipment Architecture (all the tasks performed by plant equipment). PERA therefore converted two functional architectures into three implementation architectures. All of these architectures are sub–architectures of the PERA itself. They are called architectures because they themselves form frameworks for extensive sets of tools, models, etc. for the development of their own contribution to the CIM or Enterprise program under study in carrying out the tasks involved in each case. PERA can then follows the life history of the implementation through its four remaining phases – functional design or specification; detailed design; construction and commissioning or manifestation; and finally operation to obsolescence as outlined in figure 2.2

An underlaying methodology has been created together with the Purdue Architecture. This methodology aims to provide a vehicle for explaining, organizing and guiding the development of CIM system or any type of Enterprise project based on PERA (Industry–University Consortium 1992).

PERA does not as yet require specific designated tools or techniques at each phase of the development of an integration system. This is one of its strengths. However, it could greatly aid the user if work were carried out to classify, evaluate and document the relative applicability of each of the multitude of computer–based, graphical and interactive analysis tools available. This applies to every stage of the analysis, design, construction, check–out, and operation of the integration systems. Those which are especially applicable could be formalized, if not already accomplished, for that application (Williams 1993). Some of the methods used in industrial cases of PERA include: Data Flow Diagrams, IDEF0, IDEF–Triple–Diagonal, Entity Relationships Diagrams and IDEF1.

There is no particular definition of an Integration Platforms or Integration Infrastructures, such as the CIMOSA's IIS. The Information Systems Architecture in PERA is defined as the set of computer tools used to process information within an enterprise such as computers, communication equipment, interfaces, database facilities, etc. (Williams 1991).

2.5.4. GRAI–GIM

The GRAI Integrated Methodology (GRAI–GIM) was developed by the GRAI Laboratory of the University of Bordeaux in France. This work resulted from production management studies initiated at the GRAI Laboratory as early as 1974. It has taken its current form since about 1984 (Doumeingts et al. 1986).

GRAI–GIM considers for each phase a three level model in terms of its abstraction levels i.e. conceptual, organizational (structural) and physical (operational). The conceptual level defines what to do, the organizational level defines who, where and when, and finally the physical level defines how to do it (Doumeingts et al. 1992). The conceptual model of the GRAI method has been developed from theories of complex systems and organization systems. It is comprised of two parts: the first part describes the organization of a production control system (PCS) and the second details the activities of a decision centre (DC). The production control system is split up into three subsystems: the information system, decision system and the physical system (figure 2.3). The information system is the link between the decision and physical systems. The physical subsystem is composed of machines, workers, techniques, etc. and transforms material flow. Finally the decision subsystem is split up into decision making levels, according to several criteria, each composed of one or several decisions centres (Vallespir et al. 1993).

The application of the GRAI method must be structured and conform to rigourous procedures. The method consists mainly of two phases. First, the analysis phase to analyse the current system and collect all the data necessary for designing a new system. Second, the design phase to design the system from the data collected during the previous phase by analysing the inconsistencies between the current system and the ideal system. Top-down and bottom-up analysis are applied in the modelling process. Top-down analysis is used to propagate and show the structure of the decision centres of the current system. Bottom-up analysis enables a detailed analysis of each decision centre to be performed (Regnier et al. 1993).

When applying the GRAI method two graphical tools are used: the GRAI grid and the GRAI nets. The former provides a hierarchical representation of the whole structure of the decision centres of the production management system whilst the latter describes the various activities of each decision centre (Doumeingts et al. 1993).

As the GRAI-GIM method is an decision tool, there is no Integration Platform or Integration Infrastructures defined.

2.5.5. Reference Model for Open Distributed Processing (RM-ODP)

The Reference Model for Open Distributed Processing (RM-ODP) is concerned with architecture that is intended to be applicable to most kinds of application. Other standards work items are concerned with domain-specific architectures such as: MAP/TOP (Morgan 1989), CIM-OSA (BSI DD 194: 1990), ISO-Factory Automation Model (ISO 1986), EDIFACT (BSI PD 6526:1990), etc. In a mature phase of ODP standardisation, these domain–specific architectures should each be consistent with the RM–ODP. This relationship to domain–specific architectures is illustrated in figure 2.4 (ECMA 1990). The RM–ODP provides a framework within which a growing family of ODP standards will be positioned and establishes an approach to the design of such systems which allow the construction of coherent but flexible distributed system.

The ISO/IEC JTC1/SC21/WG7 has defined five viewpoints in the RM–ODP to deal with the complexity of an distributed information system (ISO/IEC 10746–1). Each viewpoint is a model which describes the structure and behaviour of an information system in terms of a particular set of concerns. These viewpoints are:

- Enterprise Viewpoint: a viewpoint for modelling what the information system is required to do. The model from this viewpoint captures the business and administrative requirements, and policies that justify and orientate the design of the system.

- Information Viewpoint: a viewpoint for modelling the information structure of the information system. Architectures frameworks and information modelling describes the information model in terms of information structure, information flow and information manipulation constraint.

- Computation Viewpoint: a viewpoint for modelling the algorithmic structure of the information system. The operation and computational characteristics of the processes that change the information are described by the model in this viewpoint.

- Engineering Viewpoint: a viewpoint for modelling how qualitative characteristics of the information system is constructed. The model from this viewpoint is concerned with the provisions and assurance of desired characteristics such as performance, dependability and distribution transparency.

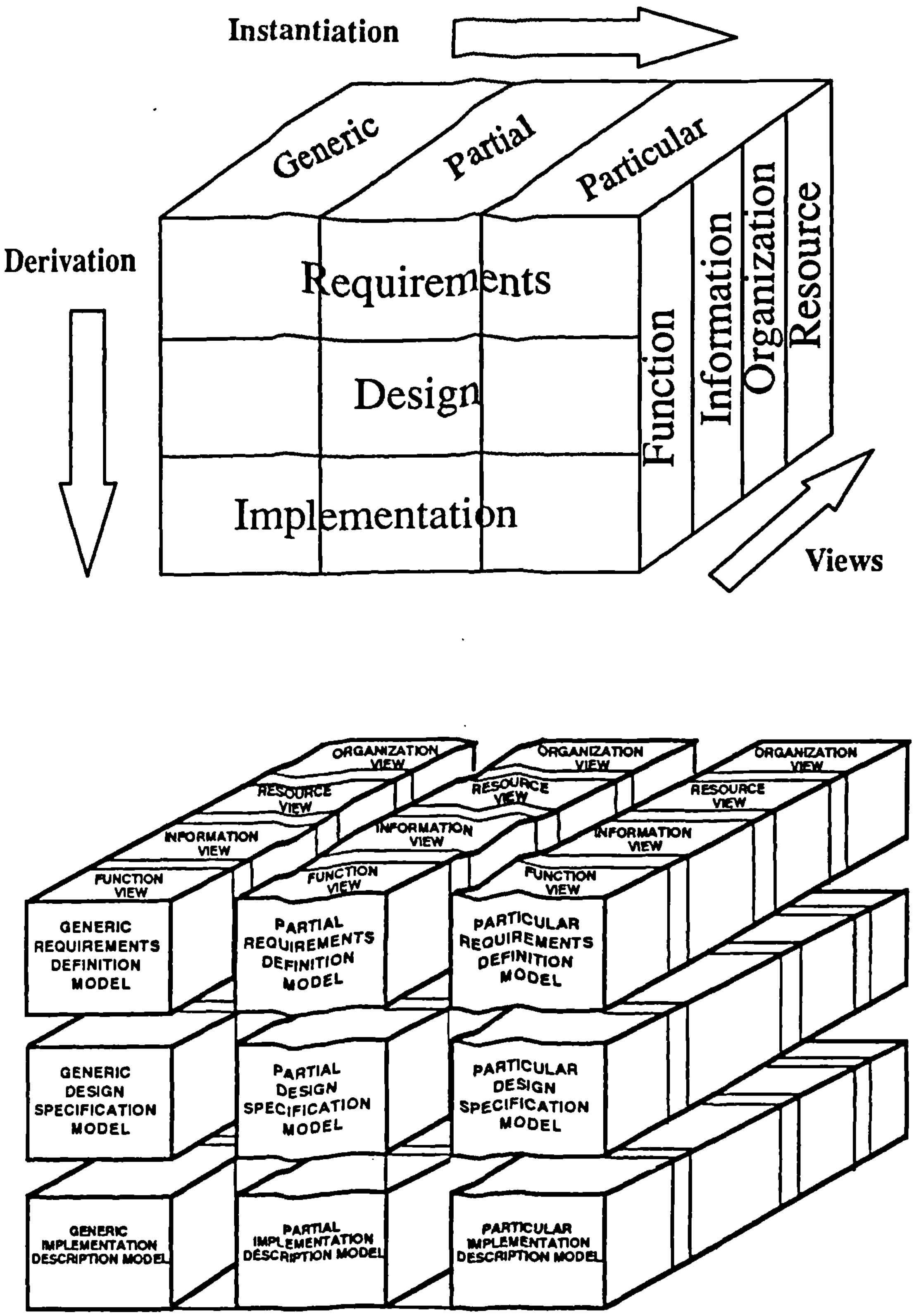
- Technology Viewpoint: a viewpoint for modelling the realized components from which is constructed in terms of computer tools, development environment and standardisation.

Each viewpoint is a complete description of the information system, concentrating on the concerns of that viewpoint. The description can be specified using formal techniques or narrative forms. A description at a high level of abstraction gives a broad overview of the system from that viewpoint. At lower level of abstraction the same system is described from the same viewpoint but with more detail.

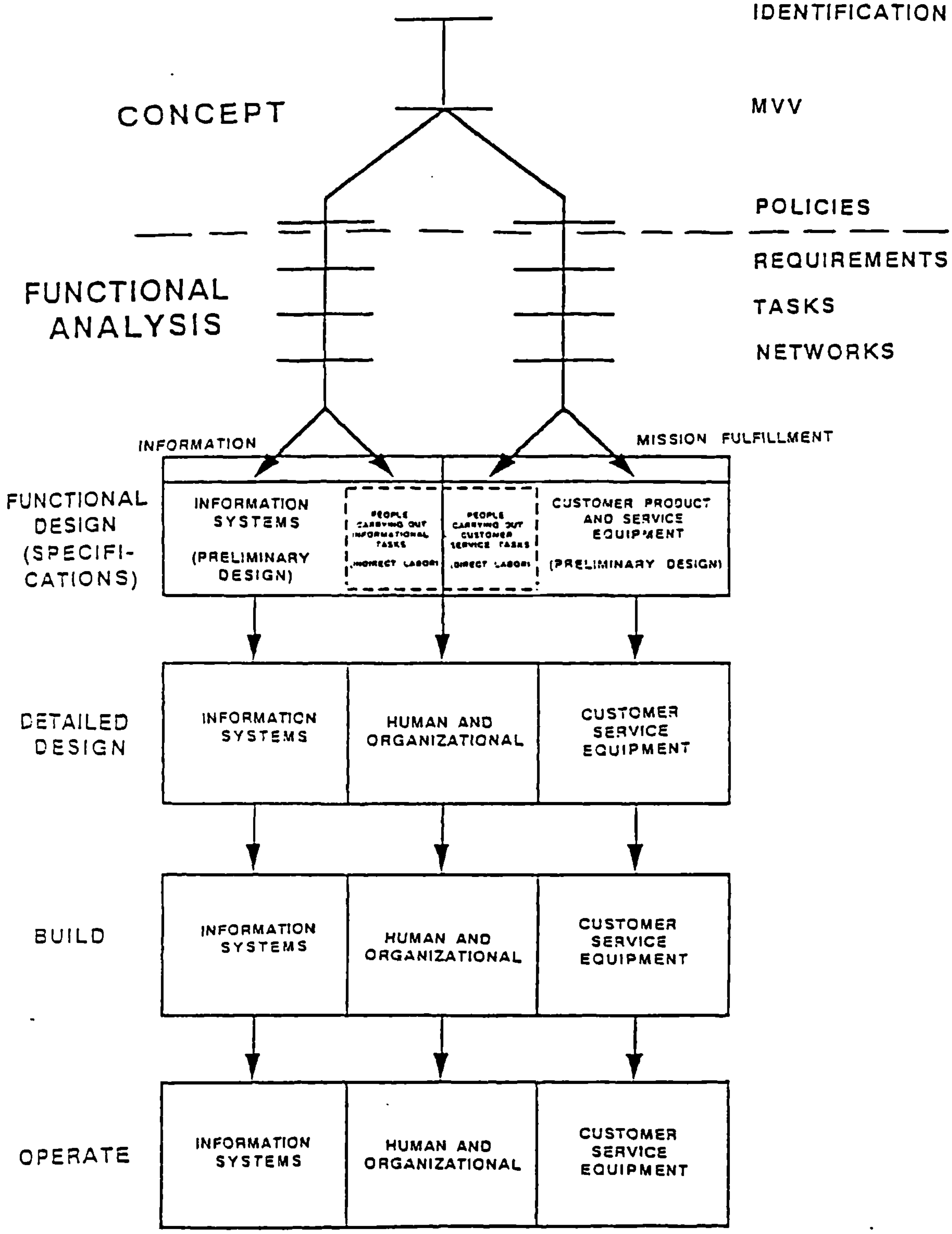
The life cycle of a system developed using the RM–ODP can be regarded as a process that may be subdivided into phases related to the different viewpoints. In this process the analysis phase deals with the enterprise viewpoint and will only consider that viewpoint and the corresponding specification of the system. The specification of a complete set of realization requirements is obtained by a process of successive refinement related to the enterprise, information and computational viewpoints. Convergence of these requirements to a specific implementation then requires choices of solution in the design and implementation phases and these choices are related to the engineering and technology viewpoints.

For each of the viewpoints, a language has been defined for writing specifications of ODP systems. The terms of each viewpoint language, and the rules applying to the use of those terms, are defined using object modelling techniques and each language has sufficient expressive power to specify an ODP function, application or policy from the corresponding viewpoint. These languages are the enterprise language, information language, computation language, engineering language and technology language. Any existing language can, in principle, be used for specification of a system from a particular viewpoint provided that those specifications can be interpreted in terms of relevant ODP viewpoints concepts.

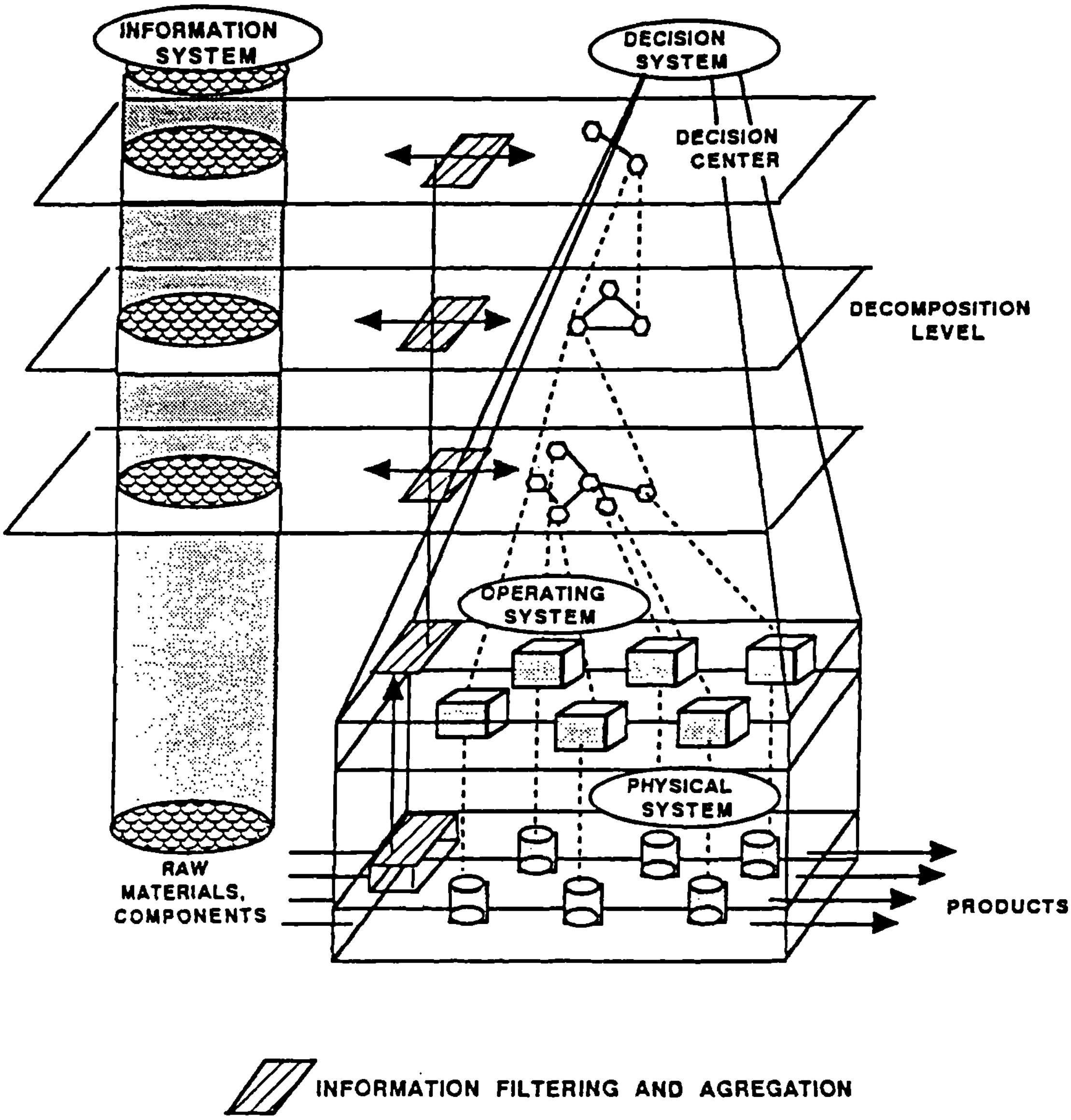
The RM–ODP does not define an specific Integration Platform or Integration Infrastructure but it does established the foundations to design and develop open and distributed information system architectures.



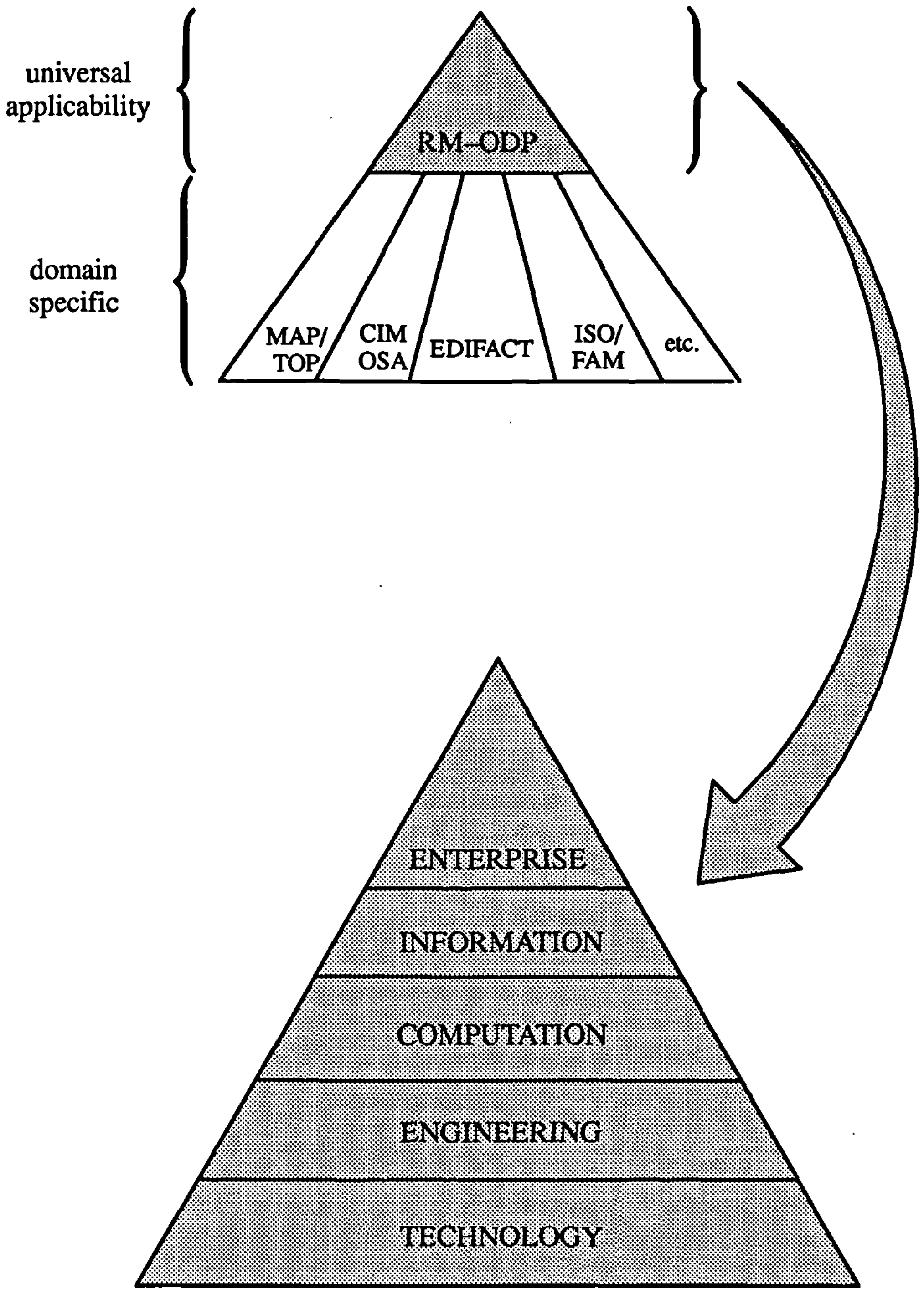
A. Molina	Figure 2.1 Overview of CIMOSA architectural framework (ESPRIT Project 688/5288)	LUT–SE Research Group



A. Molina	Figure 2.2 The Basic Purdue Enterprise Reference Architecture (Williams 1991)	LUT–SE Research Group



A. Molina	Figure 2.3 GRAI Conceptual Model (Dougmeints et al. 1992)	LUT–SE Research Group



A. Molina	Figure 2.4 Reference Model for Open Distributed Processing (ECMA 1990)	LUT–SE Research Group

Chapter 3

The Research Concept: A Manufacturing Model

3.1. Introduction

This chapter introduces the concept of Manufacturing Models. Contemporary research is reviewed to provide a reference for the Manufacturing Model research work reported in this thesis. Finally, general considerations related to the development of Manufacturing Models are defined.

3.2. Manufacturing Model Definition

The literature reviewed has shown that Manufacturing Model is a generic name which identifies two types of models:

1. CIM models which represent the business functions, software modules, control and information system architectures that can be used to design and implement CIM systems e.g. IBM Model (IBM 1989), NIST–AMRF hierarchical model (Albus et al. 1981), Siemens AG Model (Baumgarater et al. 1989), DEC Model (Flatau 1988) and Discrete Parts Manufacturing Reference Model (BSI DD 203: Part 1 : 1991).

2. Information models which represent the data and information that describes the factory in terms of either, resources and processes, or both e.g. Factory Model (IMPPACT 1991, Mantyla 1993), Facility Model (Molina et al. 1992), Manufacturing Resource Model (Kimura 1991), Manufacturing Resource Management Model (ISO TC184/SC4/WG8/N13) and Manufacturing Model (Al–Ashaab and Young 1992).

The research on CIM models has contributed to the development of new generic manufacturing models, and therefore influenced the work on enterprise modelling. One example of this work is the creation of CIMOSA models (ESPRIT Project 688/5288), which has evolved from supporting the design and implementation of CIM systems into the development of Enterprise

Models. Although the scope of a CIM model is more limited, it can be seen as a model, part of an Enterprise Model, for a specific application domain i.e. the CIM system. For a discussion of the evolution of CIM Models and a review of existing models see Rembold and NNaji (1991).

The latter Manufacturing Models (i.e. information models) have become an important element of the Enterprise Models as they basically capture and represent the information regarding the manufacturing resources of an enterprise. The development of Enterprise Models can be based on Reference models (e.g. CIMOSA, PERA, GRAI–GIM) which allow the representation of the important aspects of a company i.e. information, process, resources, etc. These aspects are captured in an Enterprise Model by modelling different views of the enterprise (Figure 3.1). One of the most important aspects in an enterprise is its information, which is modelled in the information view. Being able to capture and describe the information required to operate an enterprise is key for the realization of its activities i.e. design, manufacturing, production, etc.

The author, in agreement with other researchers, considers that the information necessary to support the realization of design and manufacturing functions in an enterprise can be captured and represented in two information models (Figure 3.2):

1. Product Model

2. Manufacturing Model

Product Model research has been carried out by academic research institutes in order to establish product modelling environments and experiment with the integration of automated applications for details see Spur et al. (1986), Kimura et al. (1987), Grabowski et al. (1989), Mantyla (1989), Kinstrey et al. (1990), Young and Bell (1992), and Bjorke and Myklebust (1992).

The rapid development of computer aided information technologies has triggered situations where a set of information models (such as manufacturing models that include tool or material flow models) have to be developed to better assist in the realization of the design and manufacturing activities (Krause et al. 1993). The emphasis of information modelling is therefore shifting to include models other than those that represent product data i.e. models which represent manufacturing facilities.

The concept of Manufacturing Models evolved from efforts of various industries and research groups to build information models for their particular applications:

- The Factory Model of IMPPACT (IMPPACT 1991, Bjorke and Myklebust 1992, Gielingh and Suhm 1993).
- Manufacturing Resources Model in a Virtual Manufacturing Environment (Kimura 1991).
- The Facility Model in a concurrent environment for FMS design (Molina et al. 1992).
- The Manufacturing Model in a design for manufacture environment for injection moulded products (Al–Ashaab and Young 1992).
- The Factory Model of the Manufacturing Cell Operator's Expert System – MCOES (Mantyla 1993).
- The standardization efforts on Resource Usage Management (ISO TC184 SC4 WG8/N13) and KCIM (DIN Technical Report No. 15).
- The CIMOSA models (ESPRIT Project 688/5288).

These Manufacturing Models only cover a limited scope of design and manufacturing activities. Nevertheless, the research results have contributed to understand better the problem of modelling manufacturing information and have arisen issues which have to be considered when developing Manufacturing Models. The following section reviews in detail the contributions of these research projects.

3.3. Contemporary Research on Manufacturing Models

Although several concepts of Manufacturing Models have been developed, their merits and structures are intensively being discussed. The aim is to develop generic models for specific type of industries, since a general–purpose generic model will be quite complex and very abstract. The present endeavours have shown that for a generic model to be practical, only the more standardized resources and operations should be modelled. An additional problem is the complexity and time demanding task to perform the information modelling work. To assist the developers in this task many ideas from software engineering have been adopted such as the use of reference models and object oriented methods. In this section a review of different Manufacturing Models is presented.

3.3.1. IMPPACT (Factory Model)

An early reference to the modelling of manufacturing facilities is to be found in the IMPPACT project documentation (IMPPACT 1991). The Esprit project 2165 – IMPPACT (Integrated Modelling of Products and Processes using Advanced Computer Technology) was developed to demonstrate a new generation of integrated modelling systems for product design and process planning including machine control data generation. The IMPPACT project was targeted to Discrete Parts Manufacturing and the following three models where defined as key elements to achieve the integration of modern CAD/CAM technologies: Product, Process and Factory models.

The Product Model in IMPPACT "covers all the information belonging to the part", the Process Model "consists of information describing the production activities as processes, operations and passes" and the Factory Model "structures the information for the facilities as for machine tools, jigs and fixtures, tools, robots, etc." The modelling languages used in the development of these models were IDEF0, NIAM and EXPRESS.

The Factory model specifies the real production system components so called production mechanisms and the physical and logical relations between them. These production mechanisms cover the technological and human resources referenced by process and operation planning and other existing means of production. There are nine types of production mechanisms: Shop Floor, Work Centre, Work Station, Tool, Tool Shop, Tool Adapter, Worker, Means of transport, Carrying Support (Figure 3.3). Any kind of production mechanism is characterized by administrative, economical and operative attributes. Economical attributes deal with acquisition cost, running cost, etc. Operative attributes deal with performance indicators, maintenance information, status information, etc. These data can be aggregated for some production mechanism such as Shop Floor, Work Centre. All this aggregated information allows production time and cost to be roughly calculated early in the life cycle of an ordered product.

All these production mechanism are related to each other. These relations define the general structure of a Factory. In IMPPACT target industries, a factory is generally decomposed into Shop Floors. A Shop Floor is decomposed into Work Centres which contains Works Stations and Workers. A Worker can be needed for a Work Station or depend on the Work Centre or in the Shop Floor. A Work Station has its own Tool Shop or used Shared Tools.

In order to be able to detect an impossible sequence of processes, there is the need to know the possible material, part or product flow between Work Centres, or between Work Stations, etc. An entity called Path specifies these possible flows, which are seen as a network into the factory.

A Work Station can either be an Automaton or a Manual Station. A Work Station can either be used as a Production Work Station, an Assembling Work Station or an Inspection Work Station. According to the application area and technology used, this model can be further extended.

The entity Means of Transport represents all kind of material handlers. It falls into two big families: dynamic and static. A Static Means of Transport carries material, parts or products without moving itself. There are several subtypes of this kind: chain through conveyor, belt conveyor, roller flight conveyor, carpet conveyor, elevator, lift, manipulator robot, etc. Dynamic Means of Transport move with the part they carry. A typical dynamic means is an Automatic Guided Vehicle. The other subtypes are Material Handling Car, Lift Truck, Carriage.

A Carrying Support retains a material, part or product during a produce, assembly, or transport phase. The subtypes of this entity are containers, pallet, etc. In case of pallet, the carrying support has its own fixture.

3.3.2. Kimura (Manufacturing Resources Model)

Kimura's (1991) Manufacturing Resources Model "tries to represent the whole factory ...... it will include the following items: machine tools for various manufacturing processes, tools, fixtures, jigs, control devices, communication equipment, materials, buildings, and human resources, etc. ".

In the virtual manufacturing environment defined by Kimura (1993a), an object model represents a virtual factory. Object models are computer executable models and are classified according to the criteria of physical constraints. The Manufacturing Resources Model consists of manufacturing resources models representing real resources that exist in factories (e.g. machine tools for various manufacturing processes, tools, fixtures, jigs, control devices, communication equipment, materials, buildings, and human resources, etc.). This model represents all the manufacturing related facts such as operational states, by which available resources at some instance can be dynamically determined. This models is to be used for factory and manufacturing equipment planning and design, manufacturability evaluation, manufacturing process/operation simulation, maintenance and system diagnosis. The Manufacturing Resources Model is a basis for Virtual Manufacturing, and part of this model has been studied as a manufacturing system simulator, but many issues have been left unsolved. Kimura (1991) states that a step–by–step development of this model is required based on the analysis of real factory resources and their operation. An object oriented approach has been used for the development of this model.

3.3.3. Molina et al. (Facility Model)

A Facility Model described in Molina et al. (1992) to "represent the production and material handling systems, and the relationships between them according to a specific Flexible Manufacturing System" is used for the design and evaluation of Flexible Manufacturing Cells in a concurrent environment for FMS Design.

The Facility model is organized in four levels of abstraction: components, structural, functional and dynamic. At the components level all the main components of a facility are defined i.e. machines, tools, material handling systems, storages, and controllers. Further sub–classification of these basic components have been developed to cover a wide range of resources such as: types of machines (drilling, milling, etc.), types of material handling systems (automated guided vehicle, conveyors, roller–tables and robots), etc (Figure 3.4) Attributes are defined for each component according with the characteristic needed, for example for a machine: machine number, name, classification, available capacity (operation space, accuracy, table load, pallet size, etc), machine costs and machine times.

At the structural level of the facility model relationships are used to define more complex groups of components to be able to define automated machine's capabilities such as a machine tool with an automated tool changer or automated pallet changer. At this level the composition of cells is possible by grouping the following components: machine, tool, material handling system, storage and controller.

The functional level holds different kinds of association links which are used to configure realistic flexible manufacturing cells. These association links represent feasible relations between machines, material handling system, controllers and storages. For example, connect two machines by using an automated guided vehicle of the adequate type. Finally, at the dynamic level, methods are programmed and attached to each cell to be able to simulate the behaviour of the cell and create a graphical animation.

The Facility Model has been developed using a combination of object oriented programming (C++) and logic oriented programming (Prolog).

3.3.4. Al–Ashaab and Young (Manufacturing Model)

A Manufacturing Model has been defined by Al–Ashaab and Young (1992) as the model that "captures the information which describes the characteristics, or behaviour, of the process and the knowledge and constraints which govern the use of the process". This Manufacturing Model represents the capabilities of the injection moulding process. The manufacturing information has been categorized into mouldability features, mould elements and injection mould machine elements (Figure 3.5). This model has been represented using EXPRESS and a design for manufacture application have been developed using the Booch object oriented methodology. The Manufacturing Model and the design for manufacture application have been implemented in the object oriented environment LOOPS.

3.3.5. Mantyla (Factory Model)

In a workshop oriented system called MCOES (Manufacturing Cell Operator's Expert System), a factory modeler allows a manufacturing engineer to store static factory resource information in the Factory Model (Mantyla 1993). The Factory Model represents static factory entities and process models. Inputs to the factory modeler are retrieved from other subsystems such as tool management system, production management system, shop control system, and informally from the factory feedback. This model holds factory specific information, reflecting the facilities and manufacturing process available in that particular factory. All data in the MCOES system has been represented as Common LISP frames.

3.3.6. WG-8 (Manufacturing Resource Management)

The scope of the ISO TC184/SC4/WG8 is to develop a standard for Manufacturing Data Exchange (MANDATE). The standard will enable an enterprise to plan, control, allocate, and manage, enterprise resources for the manufacture of a product.

Three projects have been defined in the scope of the Working Group 8:

Project 1: To develop an international standard dealing with model, form, and attributes of data exchange between an industrial manufacturing company and its environment of manufacturing management activities.

Project 2 – Resource Usage Management: To develop an international standard dealing with model, form, and attributes of data able to reside in an industrial manufacturing company's resources database to be used by manufacturing management for the purposes of managing the manufacturing company.

Project 3 – Data Controlling and Monitoring the Flow of Material: To develop an international standard dealing with the model, form, and attributes of data controlling and monitoring the flow of materials within the industrial manufacturing company from a manufacturing management viewpoint.

A conceptual model to describe the Resource Usage Management has been proposed and is being discussed within the Project 2. The model describes a resource in terms of inputs, transformations and outputs. Inputs to a resource are the product (material), preparation elements and information. Three different layers are defined to perform the transformations within a resource: transformation layer (which is often named operative layer for a machine and process layer for people), control layer (which must be mechanical, electronic or human), and information layer (which is the way to introduce information from the outside up to the control and information from the control up to the outside). The outputs are the product (material) and information which have been transformed by the resource. A third dimension of the model enables the representation of different aspects of a resource depending on which context is used, for example in their maintenance, costing, or quality management (Figure 3.6) This model is intended to be used outside the scope of manufacturing management such as in the models defined within STEP.

3.3.7. KCIM (Manufacturing Resources Model)

Major input to the standardization work of the ISO TC184/SC4/WG8 has been provided by the KCIM initiative, now called QCIM from the German Institute of Standards (DIN Technical Report No. 15). The KCIM commission has identified the development of a conceptual schema for manufacturing as a strategic area for research. Major work in this area is the development of a Manufacturing Resources Model to represent all the manufacturing resources within a company (ISO TC184/SC4/WG8 N9).

The Manufacturing Resources Model aims to provide a standardized representation of the resources structure, standardized description of limits of capacity, geometry, technology, and technical relations, and standardized data management of resources. The following breakdown of resources have been identified by KCIM: Production Resources, Organisational Resources, Supply and Disposal Units, Transportation Devices, Warehouse Resources, Measuring and Testing Resources, Furniture and Fittings.

The following classes of resources have been defined (Figure 3.7):

- A Resource Group which comprises resources that are determined by a set of identical characteristics. A group hierarchy can be built following different hierarchy criteria.

- A Generic Resource which is a resource class where the characteristics, parameters and assigned construction destinations are completely defined.

- A Specific Resource is the instance of a generic resource where all the parameters have been instantiated, except temporal characteristics such as material planing data.

- An Individual Resource is a concrete resource where individual and temporal characteristics (like material planing dates, wear conditions, etc.) can be assigned to.

The KCIM Manufacturing Resources Model will be based on already DIN standards e.g. DIN 4000/4001 to construct tables of resources characteristics and DIN 4002 for the construction of additional table of contents and a dictionary of characteristics. This model will be represented in EXPRESS.

3.3.8. CIMOSA

The ESPRIT Consortium AMICE (ESPRIT Project 688/5288) has defined the CIMOSA open system architecture (see Chapter 2, sub–section 2.5.2). Four views (function, information, resources and organization) are defined in CIMOSA to "enable the user to structure and detail specific enterprise aspects". In this framework the enterprise processes, activities and operations are modelled in the function view. The information view allows the modelling of information about products, process, resources, organization and business (administrative). The enterprise resources needed for carrying out the enterprise processes are modelled in the resource view. Finally the organization view allows the description of the responsibilities to manage processes, information and resources. The assessment of this particular project is emphasised due to its influence to the work described in Chapter 7 and 8.

The CIMOSA models are based on an object oriented modelling approach. Major classes of constructs are: Domain, Event, Domain Process, Business Process, Enterprise Activity, Functional Operation, Resource (including Functional Entities), Capability, Enterprise Object, Object View, Information Element, Authority, Responsibility, Organisation Unit and Organisation Cell. These constructs have been applied to several industrial case studies by

AMICE or by partners projects. Case studies have mainly concentrated on discrete part manufacturing.

Major constructs of the Information View are (Figure 3.8):

- Information element: an information element is any piece of information or data which can be named and, for the purpose it is being used and is indivisible. Each information element is defied by its name and its data type (integer, real, Boolean, string, array, record, date, file).

- Enterprise object: the enterprise object is a construct used to represent objects of the enterprise (e.g. machines, operator, part program, cutting tool, etc). Any object is defined by its name, its abstraction hierarchies (i.e. relations "is–a", "part–of", and "member–of") and its list of properties (e.g. for a tool : tool name, cutter type, tool length, etc.).

- Object view: users and applications of the enterprise handle/use/process objects of the enterprise. However, they never directly manipulate the objects themselves but views of them at a given instant. An object view is defined by a name, a list of properties and a set of enterprise objects. Each property is either an information element, another object view, a set of information elements or a set of like objects views.

- Object relationship: Object relationships describe user–defined, directed, links between pairs of enterprise objects. The object relationship is defined by its name, the source object , the related object, and the relationship functionality (1:1, 1:n, n:1, m:n).

- Integrity rule: integrity rules are used to express semantic rules on information elements i.e. existence of referential or consistency or validity rules constraining the range of admissible values of information elements. An integrity rule is defined by its name, its description expressed as first order logic predicates, and the list of information elements it applies to.

Two major constructs are used at the Resource View: resources and capability. A resource is a view of an enterprise object which is used in support of the execution of one or more activities and has special properties for resource management described in the Resource View. Resources are sub–divided into two major classes: passive resources and active resources. Passive resources are enterprise objects which are not capable of performing any action and are just employed (such as a tool, a probe, a cart, etc.). Active resources are called functional entities and are capable of performing functional operations on their own (such as a robot, a machine–tool, an automated guided vehicle, an operator, etc.) A functional entity can receive/process/send an even store information. Resources are also need to be categorised according to four criteria: the possibility of being moved, the possibility of being scheduled, the possibility of being replicated, and the possibility of being shared. Furthermore, resources have a location which may change over the time. Resource is a recursive construct i.e. resources can be made of resources (passive or active). Thus one can define a manufacturing cell as an aggregation of machine–tools, tools and material handling systems. Temporary aggregations of resources are called resource cells and permanent resources aggregations are called resources sets. Capabilities, i.e. technical abilities, are required by resources while functionalities of the Function View require capabilities. A capability is defined as a mapping from the set of resource to a set of values defining technical abilities. They may concern functional abilities, performance requirements, quality requirements, etc. This set is made of numeric and alphanumeric values defining characteristics such as reachability of a robot arm, capacity of a machine, capacity of a part buffer, list of skills of an employee, etc.

Essential constructs of the Function View are:

- Enterprise Events which describe solicited or unsolicited real–world happenings or requests (i.e orders) of the enterprise, which require action. Examples are the arrival of a customer order, raising a signal indicating a machine failure, sending a management order. In many cases, events carry information (e.g. the customer order, machine indications, the management order). This information is describe in the form of an object view.

- Functional operations, or simply operations, are basic units of work i.e. atoms of work (or atoms of functionality), which can be performed by functional entities (i.e active resources).

- Enterprise activities are elementary tasks i.e. pieces of functionality of an enterprise, to be performed to achieve one of the basic objectives of the enterprise (usually under constraints). Activities require allocation of time and resources for their full execution. They use function input to produce function output according to their transfer function and using their resource input. They operate under the influence of their control input and additionally they produce control output and resource output.

- Processes are recursive constructs used to model the behaviour, i.e. the flow of control of the enterprise. Processes are used to chain activities and/or sub–processes to model large business functions achieving major objectives of the enterprise under management, administrative or operational constrains and rules. Processes are triggered under some triggering conditions (involving events) and operate according to their set of procedural rules. Procedural rules are control structures relevant to CIM activities and covering sequential control, conditional control, parallelism, rendez–vous and iterative control. Control structures operate according to values of ending statuses of the processes and activities they govern.

- Domain processes and business process are user–defined functional areas in the enterprise and thus ease the modelling process. A domain is defined by its business objectives and constraints and is supposed to contain full domain process contributing to realise its objectives. A domain is thus an encapsulation of a set of domain processes receiving, sending or simply using object views and events.

The constructs of the Organisation View are: Organization Unit, Organisation Cell, Responsibility and Authority. Authority is a construct used to define the right possessed by an entity of the enterprise (usually an organisation unit) over constructs of the model. These rights are: construct creation, deletion, update and control. The Responsibility construct defines the entity of the enterprise responsible for managing constructs of the model in case of troubles at run–time. The Organisation Unit is an entity of the enterprise (person, group of persons or area of responsibility) which has authority and responsibility over some entities of the Function View, the Information View or the Resource View. Finally, the Organisation Cell is a recursive construct made of organisation cells and/or organisation units. It groups authorities and responsibilities of its constituents into decision centres.

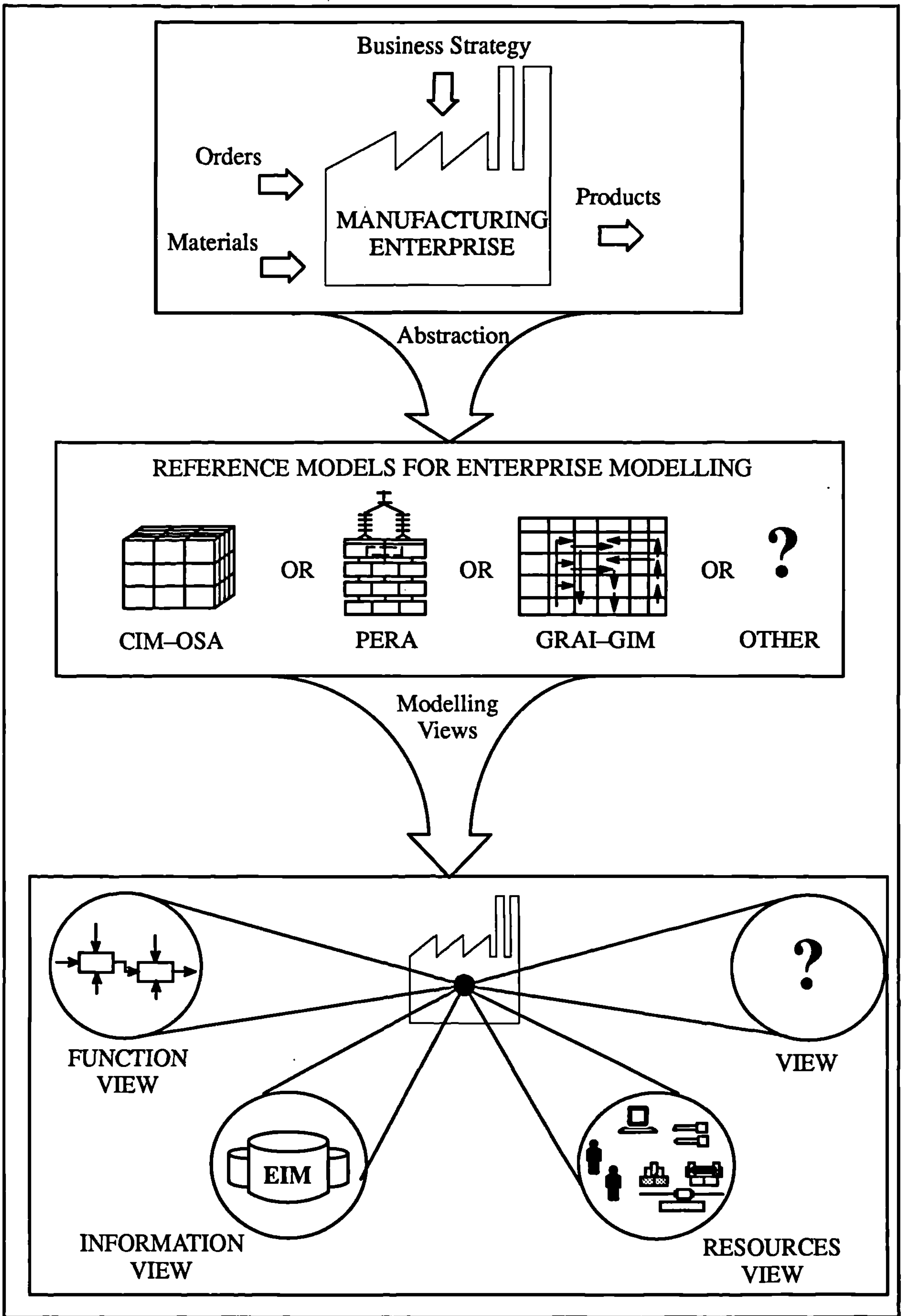
3.4. General Considerations to Develop Manufacturing Models

Since many components in these Manufacturing Models have similar functions, the conception and design of generic information structures, which allow the definition of a generic manufacturing model, is one of the major challenges in this new direction of research. However, to build a generic model which can likewise be used for different types of factories, is very difficult and may result in a high degree of abstraction. For this reason, it is very important that Manufacturing Models are well structured and very comprehensive to provide the adequate information support to the user.

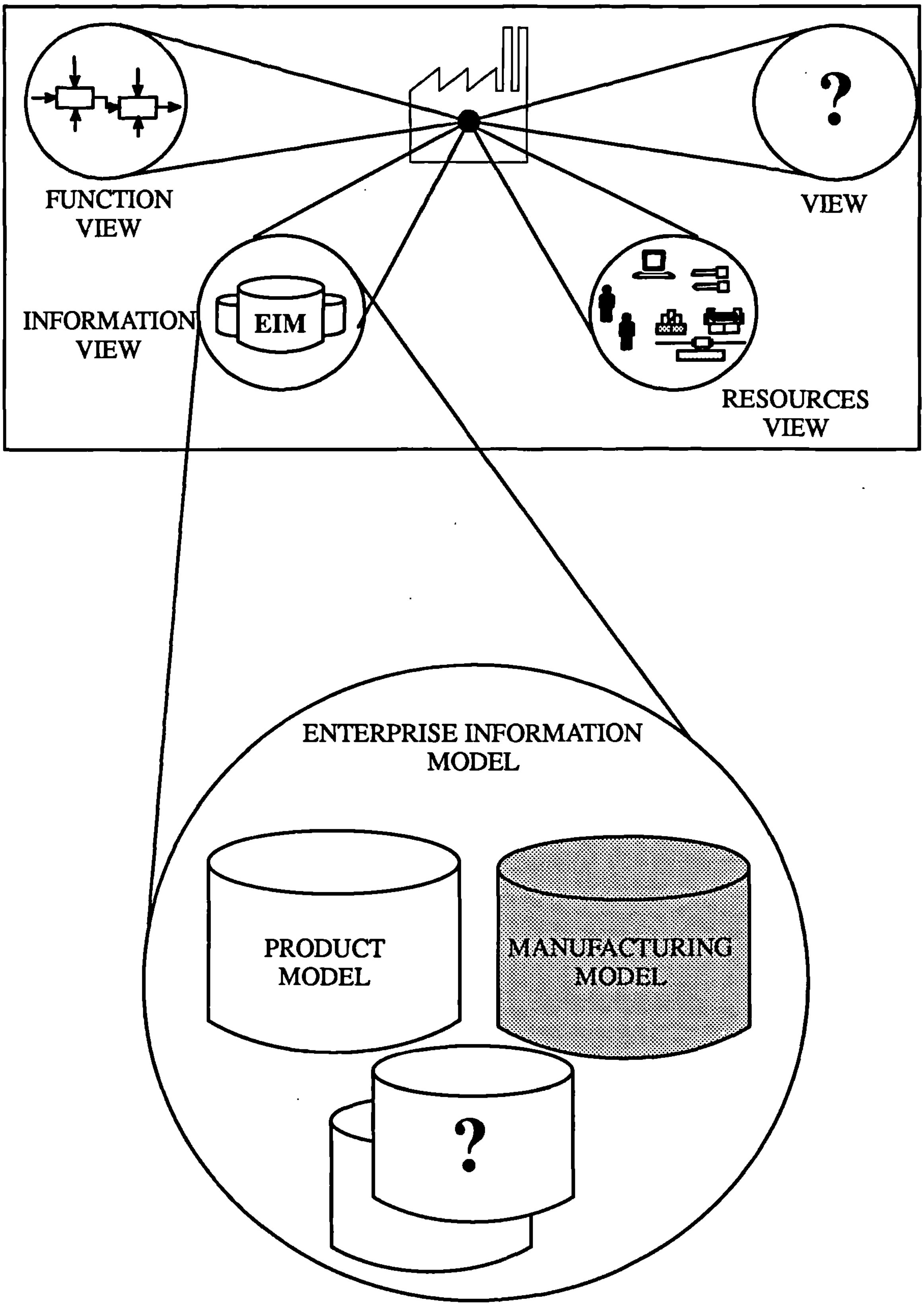
There are general considerations which must be considered when building a Manufacturing Model. As all the Manufacturing Models have their roots in early work on CIM Models, the following aspects should be taken into consideration when developing a manufacturing information model (Rembold and Nnaji 1991):

- Description of business functions
- Presentation of the material, product flow and information flow
- Representation of the manufacturing resources and processes
- Representation of the information required for planning and control functions
- Description of the interfaces and communication protocols
- Integration of the management information database
- The inclusion of time

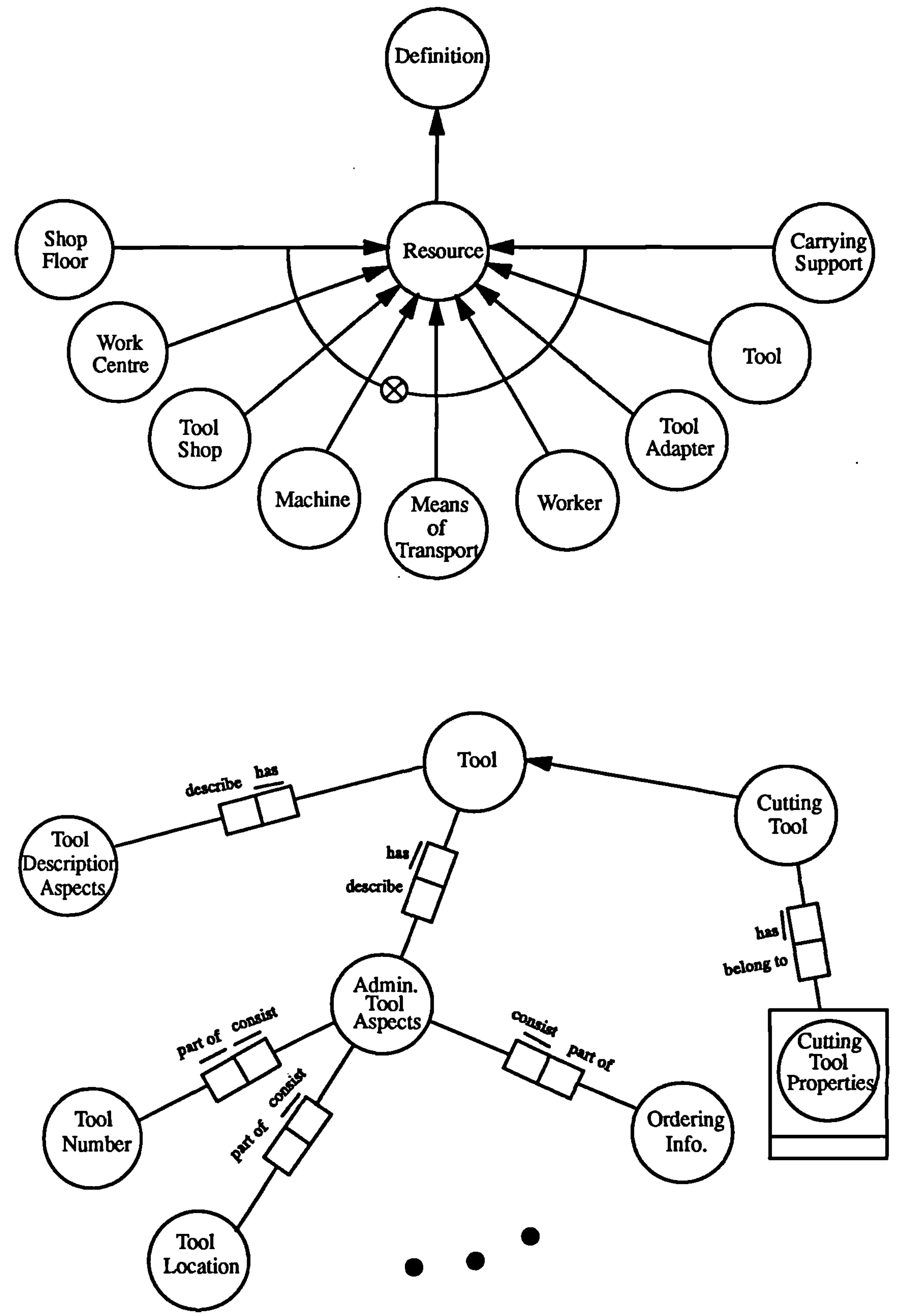
It will be very difficult to include all of them in one model, therefore the activities which are going to be supported by the model should be identified first and then, the relevant manufacturing information required by those activities should be modelled. This process will ensure that the level of detail captured in the model provides an adequate and effective support to those activities.



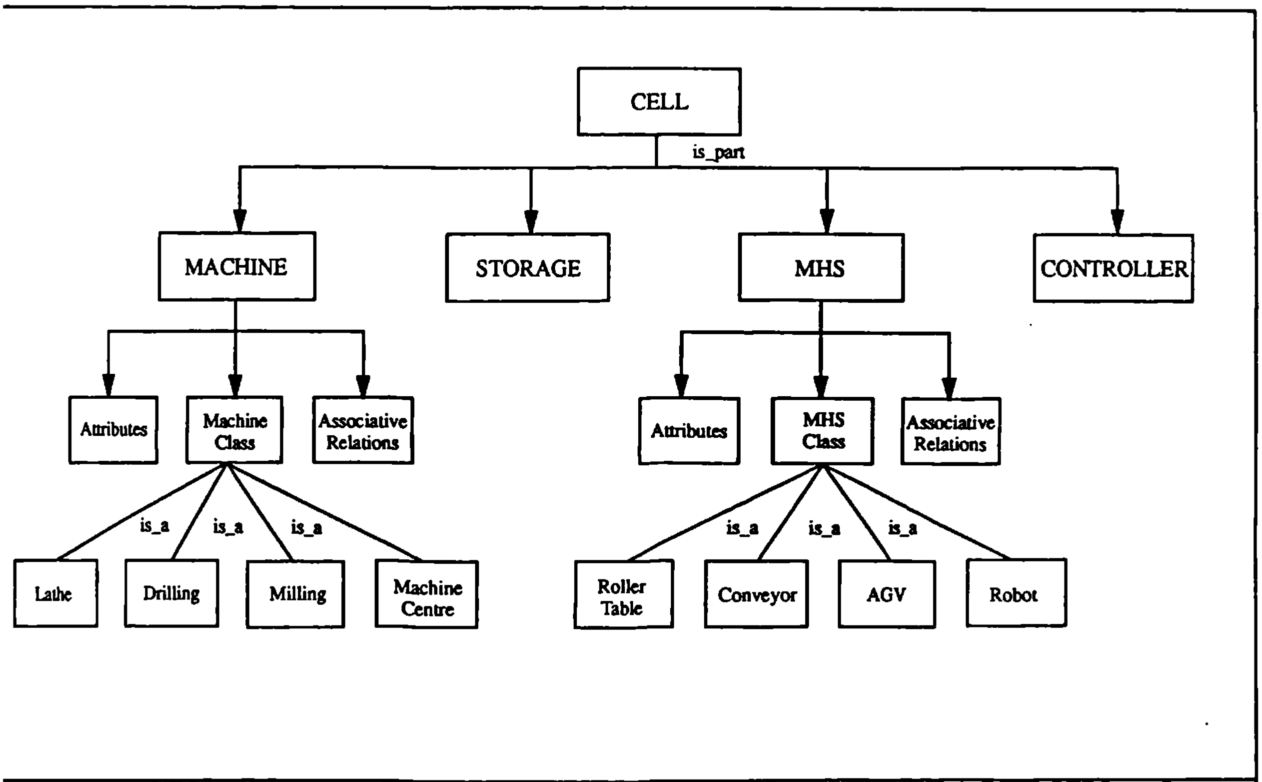
A. Molina	Figure 3.1 Reference Architecture and Enterprise Models	LUT–SE Research Group



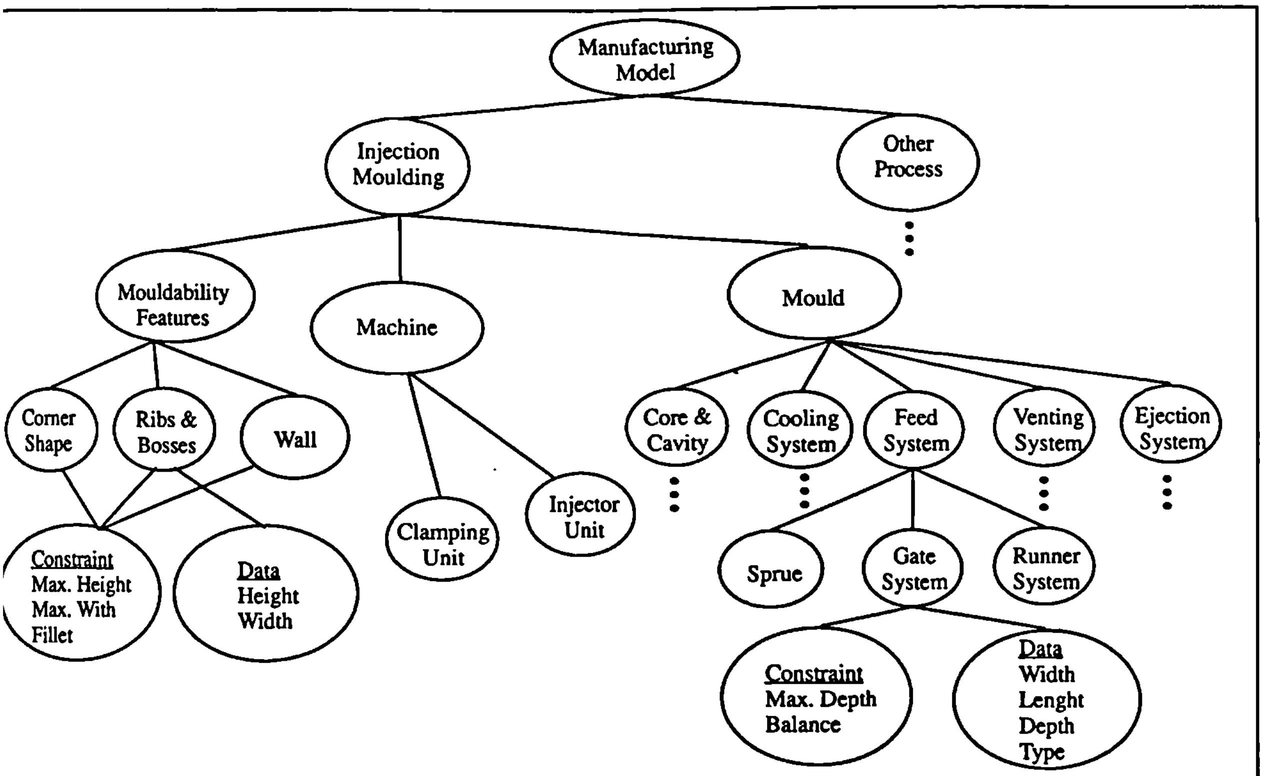
A. Molina	Figure 3.2 Manufacturing Model in the Context of an Enterprise Model	LUT–SE Research Group



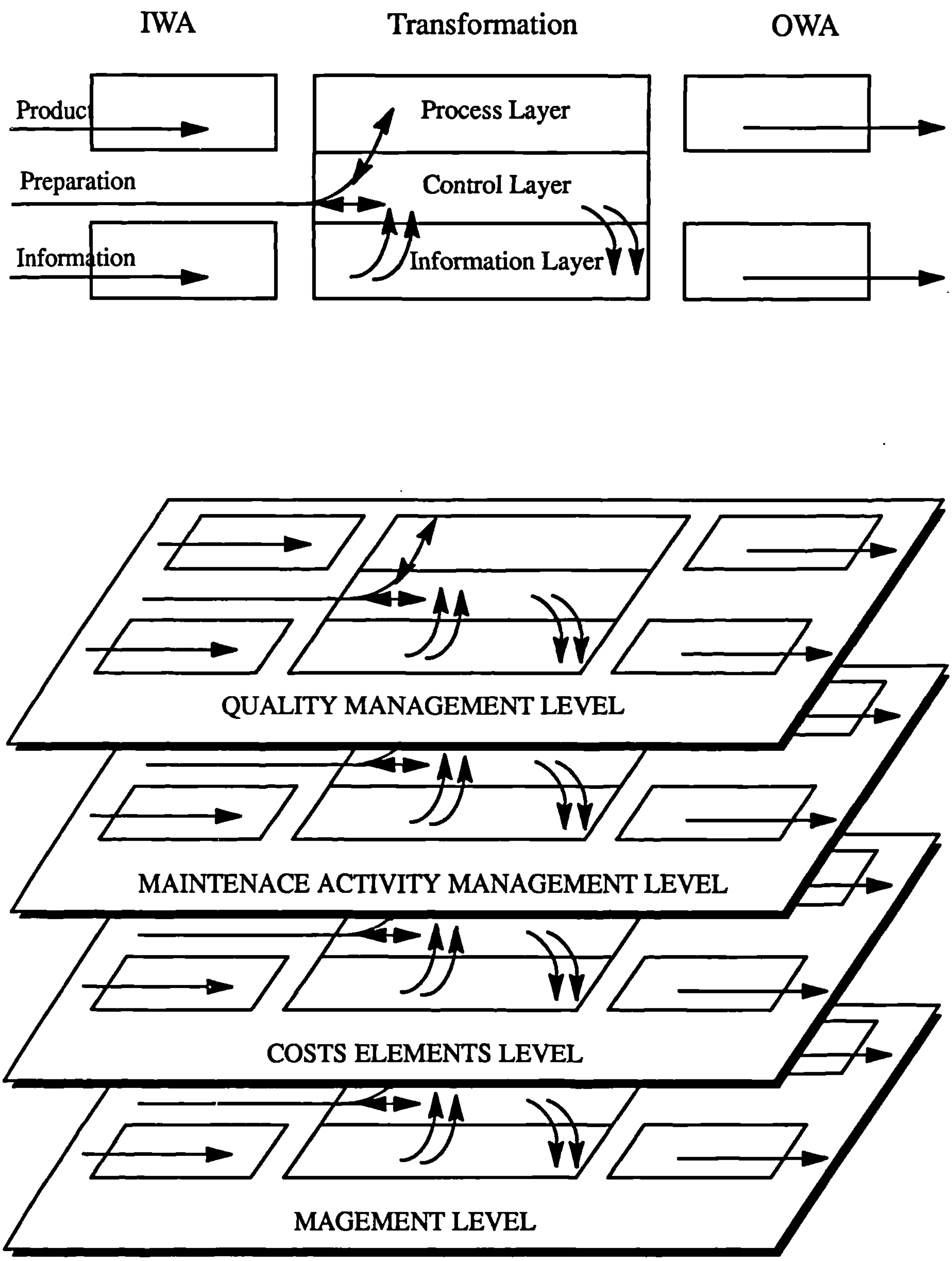
A. Molina	Figure 3.3 Example of the IMPPACT Factory Model (ESPRIT Project 2165)	LUT–SE Research Group



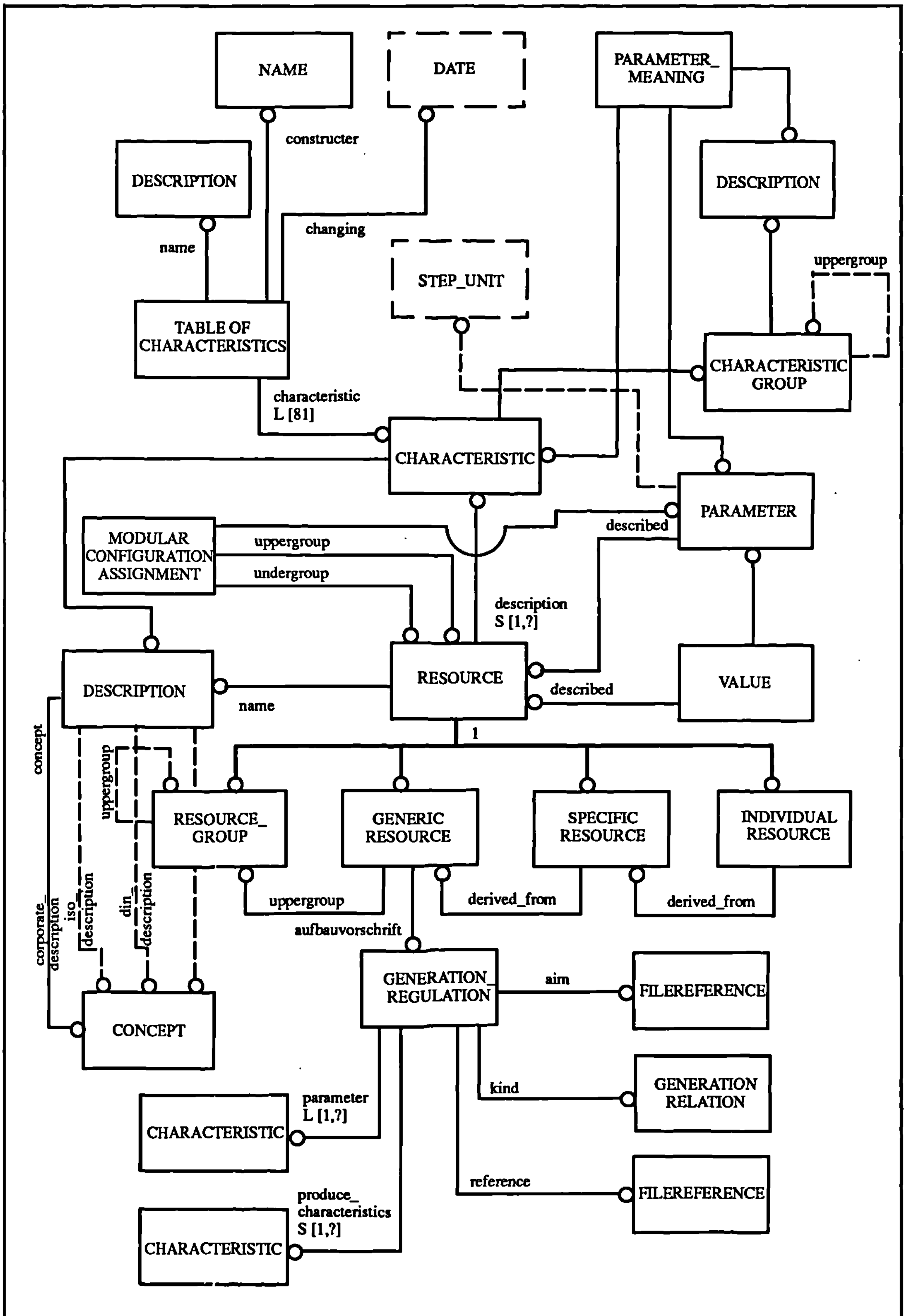
A. Molina	Figure 3.4 Facility Model (Molina et al. 1992)	LUT–SE Research Group



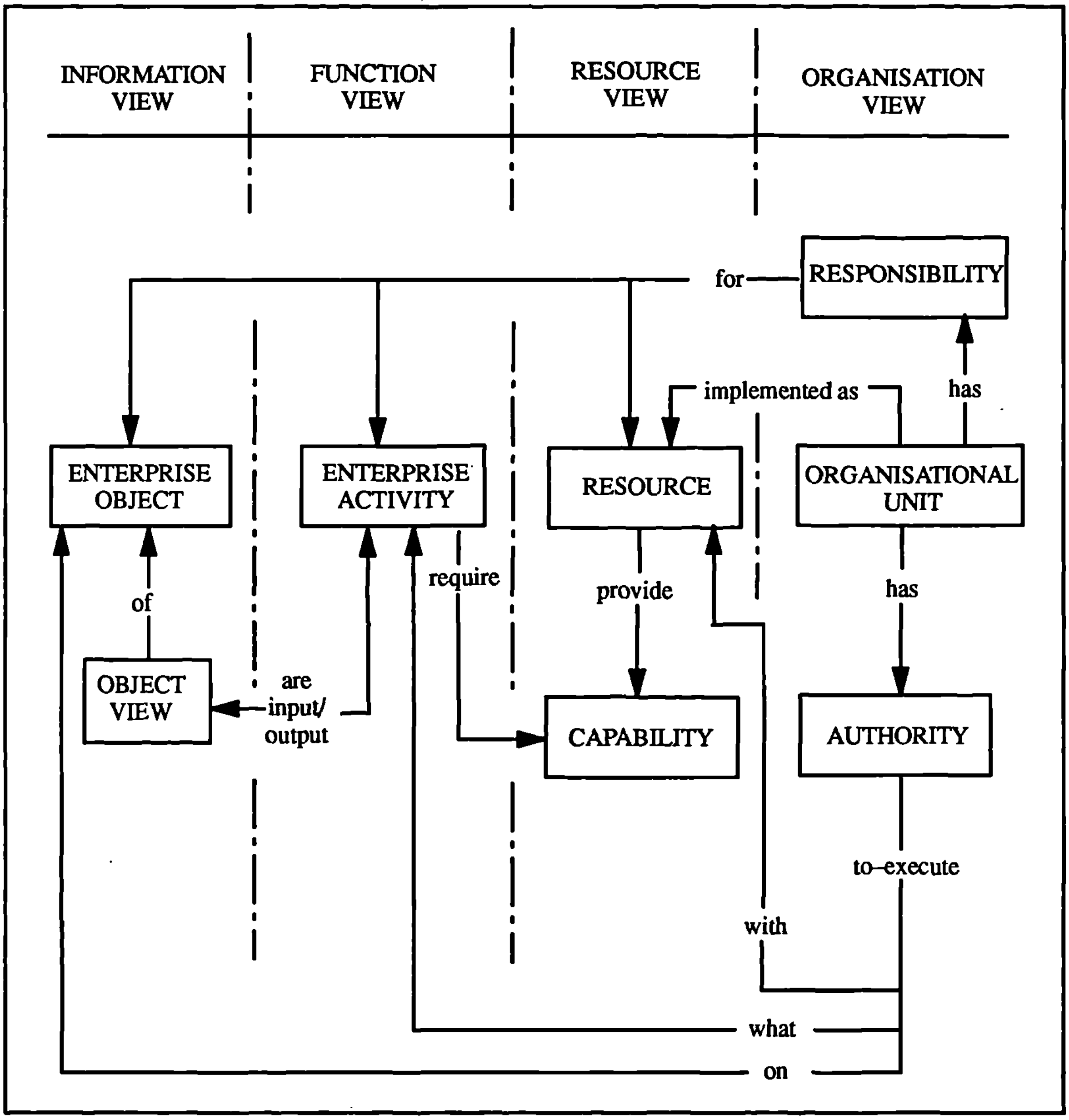
A. Molina	Figure 3.5 Manufacturing Model to represent Injection Moulding Process Capabilities (Al–Ashaab 1994)	LUT–SE Research Group



A. Molina	Figure 3.6 ISO TC184/SC4/WG8 Conceptual Model for Resource Usage Management	LUT–SE Research Group



A. Molina	Figure 3.7 KCIM Classification for Resources in EXPRESS–G (ISO TC184 SC4 WG8/N13)	LUT–SE Research Group



A. Molina	Figure 3.8 Main Relationships among CIMOSA Constructs (ESPRIT Project 688/5288)	LUT-SE Research Group

Chapter 4

The Future Context for the Manufacturing Models

4.1. Introduction

Information Integrated Systems will be required for the success of future manufacturing enterprises. The term Information Integrated Systems identifies those information systems where the primary mechanism for achieving system integration is information. The integration in these kind of systems is only possible if the information required for the successful operation of the enterprise is identified. The author's belief is that Enterprise Modelling is necessary for the development of Information Integrated System, where Information Models (e.g. Manufacturing Model) are key elements. In this chapter the concept of Enterprise Modelling is reviewed. The issues, problems and approaches in Enterprise Modelling are introduced. In addition, the concepts developed by new schools of thought are presented to outline how the issues related to Enterprise Modelling, Information Models and Information Integrated Systems are considered of relevance to pursue new concepts in manufacturing.

4.2. Enterprise Modelling: concepts, issues, and approaches

Information is the basis for successful operation of any enterprise (Scheer 1993). The information infrastructure of an enterprise is formed by the information, and all the resources which support the enterprise in its need to store, process and exchange that information. Eirich (1992) states that Enterprise Modelling is comprised of science and techniques for describing and managing the information infrastructure of an enterprise. To understand the information needs of an enterprise, it is necessary a description of the enterprise and its environment i.e. an Enterprise Model.

4.2.1. What is an Enterprise model ?

An Enterprise Model is an abstract description or representation of the enterprise (ICEIMT Workshop I 1992). Enterprise models can be regarded to be "descriptive models". These type of models intent to present a picture of what the operation of an enterprise is at a particular stage. The level of detail in a model might be from a high-level view of an enterprise to a very specific or particular enterprise area.

There are other types of models that can overlap their contents with an Enterprise model e.g. product description models, engineering design models, or financial projection models. Nevertheless these models should not be confused with what an enterprise model is. The latter aims to give a complete view of the enterprise.

4.2.2. Which are the principal characteristics of an Enterprise Model ?

An Enterprise Model must (Van Grithuysen 1992):

- represent an enterprise from multiple perspectives according to the user's needs.
- be able to answer relevant and significant questions about an enterprise.
- be accurate and complete enough to adequately support the decision making at different levels: strategic, tactical and operational.
- include access to past decision rationales, ability to survey the current enterprise-wide state, and deduction methods for formulating on going business decisions.
- be executable or be easily translated into an executable form(s)

These characteristics, if appropriated implemented in an Enterprise Model, should enable the model to be useful in supporting the realization of nowadays enterprise activities and become a really significant factor in the accomplishment of effective enterprise integration.

4.2.3. How an Enterprise model can be used in practice ?

Generally, enterprise models seem to be built for one of several purposes (Goossenaerts 1993):

- to analyse and restructure business processes for efficiency
- to realign organizational structures to better match business activities
- to design, or redesign, computer applications
- to help management have a more comprehensive understanding of their business organization
- to assist in the effective realization of business activities
- to describe and integrate office and shop floor operations in support of integrated automation

Problems that have arise with the use of Enterprise Models are (Jorgenson 1992): too complex to understand and use, difficulty to develop and maintain, the notation and representation are inappropriate to the users' view (e.g. a manager is presented to an engineering view), inadequate levels of abstraction, and their use in planning and management activities is not evident.

4.2.4. How an Enterprise model is represented ?

An enterprise model basically is composed of three main models (Eirich 1992):

1. A process (or functional) model
2. An information model
3. A resource model

The process (or functional) model to describe "processes" that create, change, combine, or destroy the entities within the enterprise. For example, engineering processes such as machining process or quality testing analysis. Related processes are often grouped together into larger categories called "functions", and individual processes may be subdivided as much as is desired for a very detailed analysis or presentation. Process models, that is, models of the processes (or functions) within an enterprise, are the core of most enterprise models. The IDEF0 notation is commonly used for this purpose, but traditional software design techniques (e.g. Yourdon, DeMarco) may also be used. New techniques such as Petri Nets have been introduced to explore the representation of more dynamic models (DiCesare et al. 1993).

Information models complement the process models. Information models provide a structured description of the information entities which exists within an enterprise, and the necessary relationships between them. Examples of enterprise entities include (Scheer 1989):

- physical objects, such as mechanical parts, buildings, machinery, material, employees, land, etc.
- information objects, such as reports, orders or computers programs
- conceptual objects, such as the ideas or engineering knowledge
- events, which indicate a change in the condition of an entity
- relationships, such as hierarchies, composite dependencies, functional dependencies, etc.

An information model conveys not only the definitions and descriptions of the entities themselves, but also their characteristics i.e their attributes. Also, by describing the relationships between or among the entities, the information model specifies some of the constraints that may apply to the existence, creation, or destruction of entities. A number of information or data modelling techniques may be used for the entity model including IDEF1X, NIAM, entity–relationship–attribute modelling, and EXPRESS, among others. Conceptual Graphs is a newer alternative. These modelling techniques, as well as some of the process oriented technologies have been reviewed in Chapter 2, Section 2.4.

The third major kind of element usually found in an enterprise model is the collection of resources needed to execute the processes. The resources include computer and manufacturing equipment, and employees. In effect, these resources are entities, but are more often represented within the process model. Nevertheless their definition is included in the information model.

Additional models can be used to create a more complete enterprise model, for example organizational or decision models. However, process, information and resource models form the core of most enterprise models in use today.

4.2.5. Issues in the development of Enterprise Models

Key questions which have to be addressed in order to cover the most important issues related to Enterprise Modelling are (Bernus and Nemes 1994):

- Which elements in the enterprise required to be modelled ?
- Who should participate in the construction of the model and what aids might be needed ?
- How should enterprise models be logically structured and organized ?
- How important is the logical consistency of an enterprise model?
- What is the meaning of completeness of an enterprise model ?
- How an enterprise model can be reusable ?

The key elements which have to be modelled while designing an Enterprise Model are: enterprise processes, enterprise information and how the processes are related to enterprise resources.

The development of an Enterprise Model requires the participation of all the users of the model. It is a necessity to involve the different users who are related to the enterprise processes to be modelled in order to have different perspectives. Especially those with the more important or significant interface roles to that particular organizational process.

CASE tools of a new type are needed to assist in the complex task of modelling the enterprise. Some characteristics of these new tools include (Bernus et al. 1994):

- Offer links to reference material and cross reference to other Enterprise Models
- Allow multiple modelling languages
- Provide the capability to translate to various representations of the same Enterprise Model
- Make the enterprise engineering process explicit and up–to–date
- Ability to navigate in the Enterprise Model via queries and links
- Ability to discover implicit properties by simulation

There are two possible approaches related to the logical structure and organization of an Enterprise Model. These being (Petrie 1992):

- Structured as one large, consistent master model, defined using a single consistent language (i.e. in much the way a computer programmer would construct a single large information system application). This is more amenable for automation to assist with building and maintaining enterprise models.
- Structured as a collection of a number of small, specialized models of different kinds, using multiple languages (i.e. more like a distributed networked information system, with different modules provided by different vendors). This is better suited for human preferences.

A series of very specialized models seems to offer significant advantages for ease of modelling, and for the fidelity of the results, as compared to a single monolithic model. There is no consensus that either contentions could always be right. It would appear that some combination of both is needed in practice, an approach that is not supported by current model building methods. The implementation of such a distributed model is also not well supported by existing distributed system technologies. In principle it is possible to have both if you have available a robust semantic structure that can bridge, and link together, some elements of each. This question seems to have hit on a significant open issue for further investigation.

Logical consistency can be or not a key issue in Enterprise Models. In single integrated models the consistency is totally important. The consistency is achieved during the process of model integration, when the one overall global representation is created. In Enterprise Models composed by multi–specialized models there is a need for a kind of unifying semantics but consistency may not be important at all. The integration is achieved by this latter approach by comparing the results of the different models needed for a particular task. Therefore the model as a whole is never fully integrated. Nevertheless in practice, there is no a real need that all the parts of an Enterprise Model be fully consistent at all times. In order to support partial integration, mechanisms can be used to record inconsistencies which will be resolved when needed. On the other hand, if there is a requirement to merge two views in a model, one can check for consistency at that time. Nevertheless the question remains: Is consistency worth the expense and effort to achieve ? The answer will vary, depending on how an enterprise model has been constructed and how it will be used.

An Enterprise Model is complete relative to the processes using the model if the processes can create (and behave according to) the intended interpretation of the model (Bernus et al. 1994). In other works, an Enterprise Model should be able to describe and provide the information required to analyse different enterprise situations according to the user who uses the model.

Bernus et al. (1994) state that the important condition of successful model reuse is that models be pragmatically complete. The factors of successful reuse are:

- Qualities of the enterprise engineer
- Reference to the type of enterprise engineering situation in which the Enterprise Model is to be used
- Ability to explicitly view the current model of the enterprise engineering process

- Quick access by the enterprise engineer to the wide range of reference material which may have used in the production of the Enterprise Model

4.2.6. Enabling Technologies for Enterprise Modelling

Several kinds of enabling technologies or improved techniques may be needed to make enterprise models really successful (Eirich 1992):

- a means to connect and integrate enterprise models with the daily activities of an enterprise.
- approaches for building enterprise models that are easy to understand and use.
- tools to lower the costs of building, maintaining, and using enterprise models.
- reliable methods for nonspecialists to update and maintain enterprise models.

The complexity of the enterprise engineering process can be significantly by reduce by standardising the following components (Fox 1992):

- Enterprise Reference Architectures
- Partial Enterprise Models (Ontologies).
- Taxonomy or categorization of business process and resources

In the future, the scope of an enterprise model might expand to become a means to actually help run the enterprise. Managers, if not most employees, might directly access an executable enterprise model to assist with decision making for certain kinds of problems.

4.3. New General Models of Manufacturing Systems

The importance of Enterprise Modelling, Information Models and Information Integrated Systems have been recognized by new schools of thought on models for manufacturing systems, namely: Agile Manufacturing, Fractal Company, Intelligent

Manufacturing Systems, Virtual Manufacturing, Virtual Manufacturing Systems, and Virtual Factory. The ways of how the above issues have been tackled varies depending on the concepts the models for manufacturing systems are based on. The following sub-sections describe each of these schools of thought.

4.3.1. Agile Manufacturing

Agile Manufacturing is a business concept. In Agile Manufacturing the aim is to develop agile properties in a manufacturing enterprise. These properties improve the company's competitive position by enabling the company to rapidly respond to changes occurring in the market environment (Iacocca Institute 1991). The agile manufacturing enterprise confers decisive competitive advantage in an open market because it is able to bring out totally new products quickly.

Agile enterprises are totally integrated organizations where the concept can be built around the synthesis of a number of agile enterprises into a single business entity called "virtual company". The ability to form virtual companies to collaborate into joint venturing operations, which are based on using each partners facilities and resources (Kidd 1994a).

Lean manufacturing and flexible manufacturing are different concepts to Agile Manufacturing (Sheridan 1993). Lean manufacturing is concerned with doing everything with less i.e. in less time, with less resources, etc.. Flexibility implies adaptability and versatility. Agile manufacturing assimilates the full range of flexible manufacturing technologies along with the characteristic of lean manufacturing. Agility is defined in terms of being dynamic, active and ready to react. Agility requires integrating flexible technologies of production with the skill base of a knowledgeable workforce, and with flexible management structures that stimulate cooperative initiatives within and between firms (Iacocca Institute 1991). Therefore agile manufacturing is accomplished by integrating three resources – workforce, management and technology – into a coordinated, interdependent, system.

A key factor to achieve Agile Manufacturing is to lever the skills and knowledge of the workforce. A knowledgeable workforce, expected to display initiative and assisted with the technology to exercise it, it is the single greatest asset of an agile enterprise (Kidd 1994b).

The ability to use and exploit the agile properties requires that managers understand and fully utilize the enterprise resources. Techniques have to be learned and developed for managing companies that promote workforce initiative at the operation level, as well as performance measures for self–directed, inter–enterprise, project teams. The guiding principle of agile enterprise management is not automatic recourse to self–directed work teams or to virtual companies, but full utilization of corporate assets (Iacocca Institute 1991).

There is a need to use advanced technologies to achieve Agile Manufacturing. However, technology has to be seen as the means to accomplish a comprehensive set of enterprise goals subjective to a managerial decision–making process. One of the most important characteristic of the agile enterprise is "Total Enterprise Integration" (Iacocca Institute 1991, volume 2, page 7). This characteristic, together with others, is supported by nine manufacturing elements, these being: business metrics and procedures, communication and information technologies, cooperation and teaming factors, enterprise flexibility elements, enterprise–wide concurrency technologies, environmental enhancement, human elements, subcontractor and supplier support elements, and technology deployment elements. The elements of the manufacturing enterprise which support the enterprise characteristics are made possible by enabling subsystems where the main critical component is "Enterprise Integration" (Iacocca Institute 1991, volume 2, page 35–37).

The following characteristic for the enterprise integration system have been defined in order to facilitate flow of information and coordinated decisions and actions in the agile enterprise:

- Effective communication of information for all domains based on an Enterprise Integration Architecture
- Information access to integrated sources of data in order to satisfy the information needs.
- Monitoring and automatic tracking of the evolution of decisions and events
- Support of cooperative work to allow team members to share knowledge and information
- Integration of independently develop software packages and systems which can be easily used and maintained.

A set of critical components have been identified as enabling mechanisms to achieve an enterprise integration system. These being:

- Standard Enterprise Integration Architectures
- Evolving Standards for Product Representations
- Heterogeneous Distributed Databases
- Evolving Standard User Interfaces
- Unified Training
- Information Services
- Computers and Networks
- Encryption, security, reliability and integrity

These critical components highlight main issues which require further developments such as the achievement of a standard information system architecture for enterprise integration, the definition of standard information models not only for product representation but for knowledge about the production process and manufacturing facilities, the implementation of new technologies to create open, distributed, reliable information systems, and last but no least, the introduction of training programs for the use and management of these new information technologies.

4.3.2. Fractal Company – Fractal Factory

Fractal Company or Fractal Factory is a term given by Professor Warnecke (1993) to a new concept for the factory *with* a future. The Fractal Factory is an integrating approach where the basic idea is the creation of self–regulating organizational working groups (fractals), each within its own area of competence. The coordination of the input and output values of the fractal is achieved by means of the superimposition of a computer assisted information and communication system. The Fractal Factory is characterized to be :

- Self–similarity
- Self–organization
- Dynamic and Vitality

The characteristic of self–similarity refers primarily to the goals of the company and its fractals. The company's goals, like those of the fractals, are self–similar. They differ in detail and specific implementation but as the goals of all fractals are similar the possibility for an coherent integration is feasible (Warnecke 1993, pages 139–142). All support mechanisms concerning the organization are available to all the fractals, especially, the information.

Self–organization in the Fractal Company affects operative, tactical and strategic levels. Operative self–organization means the application of suitable methods for controlling processes e.g. production control. Self–organization at the tactical and strategic level applies in order to achieve global objectives locally i.e. the application of adequate methods for achieving tactical and strategic plans by each of the fractals (Warnecke 1993, pages 142–145).

In the Fractal Factory cooperation between the self–regulating and self–organizing factory fractals is characterized by high individual dynamics and maximum ability to react to dynamically changing conditions. This is called process dynamic structuring. Vitality means constantly discovering and taking advantage of success factors, internal and external.

Fractals are networked via an efficient information and communication system. Therefore there is a need to pursue a comprehensive modelling of data and functions where all the static and dynamic interrelations of a factory are integrated in order to make them available to an optimized computer aided system (Warnecke 1993, page 161).

In a Fractal Factory, therefore, particular significance must be attached to the following points (Warnecke 1993, page 162):

- Model language paradigms e.g. object–oriented and agent concepts which support the synthetic aspects of fractal thinking
- user openness and transparency in CIM systems
- expert–system supported information gathering and compressing
- provision for evaluation via simulation prior to the execution of expensive operations
- knowledge based process scheduling, execution and control systems
- intelligent control mechanisms providing short feedback loops between decision maker and real process

If CIM is regarded as an information and communication tool, CIM systems must provide flexible and efficient information and navigation systems for the fractals of a factory. Factories are therefore information processing systems, in this scenario manufacturing can be seen as the conversion of information. The amount of information processing determines the manufacturing structure, and hence the competitiveness of the factory. In the fractal factory concept the information is a resource, which has to be managed and rationalized. In this context the amount of information processing required to perform a manufacturing task should be kept to a minimum. Information systems have the task of providing the data required for the manufacture of products and the allocation of operating resources within the framework of a suitable manufacturing processes.

In order to facilitate the integration of fractals there is a need to define a universal data model i.e. a universal information model (Warnecke 1993, page 113). This model has to be supported by suitable data structures, along with effective communication channels.

Warnecke (Warnecke 1993, page 219) states that the fractal factory concept is "food for thought and an aid to orientation".

4.3.3. Intelligent Manufacturing Systems

The Intelligent Manufacturing Systems (IMS) is a type of next generation of manufacturing systems in which the whole manufacturing process, from order booking, all the way through research and development, design, manufacturing, distribution and management, is realized through autonomous production systems comprising of equally intelligent components. These components are integrated at a high level in order to be able to cooperate in an intelligent manner (Yoshikawa 1994).

An IMS program was proposed by Japan in 1989 to explore the possibilities for international cooperation in advanced manufacturing. Under the mandate of the IMS International Steering Committee (ISC), international discussion took place for the first time on the technical aspects of a feasibility study. The ISC identified six technical themes as being appropriated for the test cases during the feasibility study: enterprise integration, global manufacturing, system component technologies, clean manufacturing, human and organisational aspects and advanced materials processing.

The test case which included the theme of Enterprise Integration was Globeman 21 (Global Manufacturing in the Twenty–first Century). The purpose of this study was to establish criteria for facilitating efficient global manufacturing businesses. Topics include concurrent engineering, inter–enterprise management (including supply chains involving different time zones and enterprises), and enterprise integration. The technical theme of Enterprise Integration included topics such as modelling, system architectures, network systems, database technologies, product design technologies and product data exchange technologies.

4.3.4. Virtual Manufacturing, Virtual Manufacturing Systems and Virtual Factory

The modelling and simulation of all the necessary manufacturing activities for product realization by computer support technology is called Virtual Manufacturing (Kimura 1991). The Virtual Manufacturing approach proposes the following:

- systematic organization of manufacturing knowledge.
- comprehensive modelling of engineering objects and activities.
- precise computer simulation before making real manufacturing to evaluate design and manufacturing activities results
- elimination of inappropriate results generated by the computer simulation evaluation.
- maintenance of models in daily operation to achieve high–quality simulation.

Product and Process Modelling methods are considered the kernel to realize the Virtual Manufacturing concept. These modelling techniques are essential to represent the structure and evaluate the behaviour of products and associated manufacturing processes. In addition, Activity Modelling is necessary to represent various activities, whether human or by computer, for product and production engineering. Nevertheless, so far activity models have not yet deeply investigated (Kimura 1993a).

In Virtual Manufacturing a product model is a generic model used for representing all types of artifacts which appear in the course of manufacturing. It represents target products, their materials and intermediate product, tools and machines, and any other manufacturing resources and environmental objects. Process models are used to represent all the physical processes which are required for representing product behaviour and manufacturing processes. The relation between product and process models is based on reference data or data structures defined in the product models. Whenever a process model is required to describe a property about an artifact, the suitable process model is triggered automatically or with the human intervention. Processes models can be very complicated, therefore different types of methods are required for their representation i.e. declarative, procedural, and computational methods (Kimura 1993b).

Virtual Manufacturing allows easy reconfiguration and extension of a system by offering systematic and modularized knowledge of manufacturing. Total manufacturing integration by using the Virtual Manufacturing concept defines an environment where product realization activities are first performed with respect to the virtual world, where all the necessary manufacturing artifacts and activities are modelled by using activity and product/processes models. Those models includes various engineering activities (e.g. product design, manufacturing preparation, and production management), corresponding models of target products, their materials and intermediate products (i.e. Product Model), and manufacturing resources (i.e. Manufacturing Resource Model). Sometime in parallel with, and primarily after virtual activities, real manufacturing operations are performed with respect to the real world. Always comparison of models with reality and various model maintenance operations are necessary.

Onosato and Iwata (1993) have developed the concept of a Virtual Manufacturing System. According to them every manufacturing system is decomposed into two different subsystems: a "Real and Physical System – RPS" i.e material, products, machines etc., and a "Real and Information System – RIS" i.e information processing activities and decision making activities. Using these definitions as a basis, a Virtual Manufacturing System is composed of:

- a computer system which simulates the responses of a RPS called "Virtual and Physical System–VPS", and
- a computer system which simulates a RIS and generates control commands for the RPS called "Virtual and Informational System".

The Virtual and Physical System consists of a Factory model, Product Models, and Production Process Models. The Production Process Models are used to determine the interactions between the Factory Model and each instance of Product Models, and appropriate CAE systems. A Factory Model and Product Models are not static and should cover the factory life cycle and the product life cycle respectively. The Virtual and Informational System includes an Activity Process Model which describes decision making processes and the flow of information in the VIS, and the simulation engine which interprets the process descriptions and executes the decision making process (e.g. designing, planning, scheduling, controlling etc.). Based on this architecture for Virtual Manufacturing Systems some applications have been developed for factory modelling, product life cycle modelling, manufacturing process modelling and time information modelling (Onosato and Iwata 1992, Onosato et al. 1992). It is believed that the Virtual Manufacturing System can be used in a diversity class of manufacturing application i.e. designing of shop floor layout, estimating control strategies, scheduling, simulating operations int the factory, and so on. On the other hand, the development of Virtual Manufacturing Systems can provide a framework for integration of information systems due to the fact it is based on a number of enabling concepts such as information models, process models and activity models.

Virtual Factory is a new keyword to master the dynamic changes of products, product components, production processes and sales volume and to overcome problems resulting from manufacturing processes in different countries to reduce cost. In this concept divisions of different enterprises are grouped together to form teams for marketing activities, research, development, production, and maintenance (Rembold et al. 1994). Such a structure can changed easily and dynamically if there is a need for it. With the increasing capabilities resulting from the telecommunication technologies such as data highways together with virtual reality systems, the direct cooperation among teams of different companies will be possible (Biocca 1992). Compiling of several units of different companies to a new one must support the flow of information, the flow of material, the hierarchy control, and the dependencies among different separated units during the product life cycle (Gorason 1992). It is possible to distinguish two cases of virtual factory problems. The simpler case is the compilation of a virtual factory from units that are all part of the same company. The second case is the compilation using units of different companies that are perhaps also competitors for other products or services.

To compile such a virtual factory, software tools are required to support this concept of cooperation among units of one or more real companies. This software tools are:

- Distributed Computer Networks to allow data exchange among units.
- Distributed Data/Knowledge Bases to share data and knowledge for cooperative design
- Cooperative Design Systems to make use of concurrent engineering methods that are required in virtual factories.
- Virtual Reality Environment to support the man, who works in the virtual factory, with an adequate user interface. It also should graphically represent the different dependencies in the virtual factory and its hierarchical structures.
- Methodologies to develop models which describe adequately companies, factories and units.

The main problem in the construction of Virtual Factory models is to describe adequately:

- Properties of one unit
- Goals and behaviour of a unit
- Input / outputs of units
- Type of cooperation between units, factories and companies

Furthermore, software tools for the Computer Integrated Factory Design are required for a fast and appropriate reaction to changing constraints and dependencies. First steps towards virtual factory models considering different problems arising with this idea have been made in (Onosato and Iwata 1992, Lee 1993, Tsukune et al. 1993). In these type of scenarios Manufacturing Models will play a key role in successfully representing one of the major assess of a company, i.e. its manufacturing information and knowledge.

Chapter 5

The Context for the Manufacturing Model Research

5.1. Introduction

The MOSES research concept is described in this chapter to set the context, requirements and scope for the Manufacturing Model research.

5.2. The MOSES Research Concept

The MOSES (Model Oriented Simultaneous Engineering System) research has its roots in an SERC funded project that investigated an architecture for Information Support Systems. This project, undertaken jointly by Loughborough University of Technology and Leeds University, concentrated on developing structures for a Product Model that would be capable of representing more than the geometric elements of a product. The project also looked at how the information within this model could be used by manufacturing applications in order to generate process plans for prismatic parts. The outcome of the work was a contribution to the evolving STEP standard, a greater understanding of the need for a single source of manufacturing information, and a requirement to investigate architectures more suitable for supporting a diverse range of applications and data models (Corrigall et al. 1992, Young and Bell 1992).

The results of this project encouraged the SERC to fund the MOSES research project, which was again a joint undertaking between Loughborough and Leeds Universities. The specification of the research into MOSES focused on a computer based system (CAE System) that provides product and manufacturing information, enables decision support based on these information sources and is co–ordinated in a manner that makes it suitable for operation in a simultaneous engineering environment (ACME/SERC 1991).

CAE systems are highly complex and the activities of co–ordinating and structuring their design and development necessitates the use of formal methods for their representation.

Therefore a key issue in the MOSES research is the development of a CAE Reference Model to provide a framework to support the development of existing and new CAE systems by establishing a generic set of viewpoints. Methodologies and tools are recommended to assist in the specification, development and analysis of each viewpoint (e.g. IDEF0, EXPRESS, Booch). This will ensure that certain key issues are considered during the design of a system, and that standardised methods are used for the design and documentation of the system. A combination of reference models, methodologies and computer tools have been defined by MOSES, to enable the achievement of these important issues, in a CAE Reference Model.

The MOSES CAE System concept is shown in figure 5.1 and consists of two information models (Product Model and Manufacturing Model) linked, by an integration environment, to a number of Application Environments i.e. Design for Function, Design for Manufacture, etc. The operation of MOSES is such that any number of Applications Environments may be supported. Application environments are a set of data–driven applications and software tools used to provide computer support to a particular activity of the life cycle of a product. These Application Environments are sometimes referred as 'Design For X' applications. The configuration and functions of the selected application environments will depend on the needs of the host organisation e.g. a 'design for maintenance' application environment may be important to an automobile manufacture but not required by a disposable watch manufacturer. All product related information is stored within the Product Model as a design evolves. This is the sole consistent source of product information. Should an application be triggered, then it operates on product information from the product model and any product information that it generates is added to that model. The Manufacturing Model is the sole source for manufacturing information and hence all applications obtain their manufacturing information from it.

In developing the MOSES system it is intended that the information and procedures necessary to support that part of the simultaneous engineering process concerned with concurrent design for manufacture be identified, and that a prototype knowledge and software environment be used to demonstrate this. To this end an application environment for the support of design for manufacture is being developed (Ellis 1995). This makes extensive use of both product and manufacturing data and hence exercises and tests the validity of the information models.

In order to ensure industrial relevance a total of eight collaborating organisations contribute to the project. The collaborators represent a cross section of industrial sectors but all are involved in the research because they are both able to contribute valuable input to the project and also benefit from the learning generated by it. They include large manufacturing organisations, software houses, computer vendors and integration consultancies.

5.3. Requirements of the Manufacturing Model in the MOSES research project

The Product Model captures and structures the information of a product and its components through its life cycle. In the MOSES concept, along with this model it was identified the need for a new information model to describe and capture the information about manufacturing resources and processes required for the product realization, such as machines, tools, fixtures, machining cells, operators, etc., and how these resources and process are organised and used in a company. This information model has been named: Manufacturing Model.

In order to provide reliable and consistent manufacturing information the Manufacturing Model requires to describe the manufacturing facility of a particular enterprise. Especial attention is giving in this research work to the description of flexible manufacturing systems. To achieve general applicability, the model must be comprised of the entities that are relevant and important for any type of manufacturing firm, such as: manufacturing resources, processes and strategies. These manufacturing entities has to be organized in functional levels to achieve a generic representation suitable to types of facilities. One major characteristic of the Manufacturing Model is that the description of the manufacturing capability of a firm is product independent, therefore the model can be used to support the design and manufacturing activities of all products within a company.

The Manufacturing Model is the source of manufacturing information for design and manufacturing applications of the MOSES research project. It allows the sharing of common data between a diverse range of design teams and software applications in the Design for Manufacture Environment. While the outline structure for the Manufacturing Model allows for the incorporation of the full range of manufacturing information, the MOSES research project concentrates on the information structures to support a design for manufacture application, called the Manufacturing Strategist, which addresses the machining and injection moulding processes (Figure 5.2). An application called the Engineering Moderator ensures that the evolving product design considers the different life cycle activities that are represented by the application environments. The Moderator monitors the product model to ensure that conflicts in information requirements are identified (Harding and Popplewell 1944).

5.4. The Scope of the Manufacturing Model

The Manufacturing Model research undertaken in this thesis has been influenced by the concepts developed by the research projects reported in Chapter 3, particularly on the results of IMPPACT, KCIM and CIMOSA. The Manufacturing Model has been defined in this research as:

"An information model, which identifies, represents and captures the data, information and knowledge, that describes the manufacturing resources, processes, and strategies of a particular enterprise. This enables the provision of the necessary manufacturing information for the support of the manufacturing decision making in concurrent design of products".

The author's main objective is to contribute with generic solutions to the problem of capturing and representing manufacturing capability information in order to support the realization of concurrent design of products. Therefore, the Manufacturing Model supports the applications defined in the Design for Manufacture Environment of the MOSES research project (i.e. Manufacturing Strategist, Machining application and Injection Moulding application) by providing consistent and adequate manufacturing information (Figure 5.2).

The Manufacturing Strategist requires the following types of information from the Manufacturing Model (Ellis 1995):

- Information on resources e.g. resource capability in terms of processes, forms, tolerances, cost, setup, consumables and administrative information.
- Information on processes e.g. tolerance bands, start up lead time and costs, production lead time and costs, form suitability, material suitability, pre and post process implications.
- Strategies on whether to optimise for cost, lead–time, quality or some combination of these.

The manufacturing information required to support the Design for Machining application is concerned with:

- Type of resources that are available in a particular factory, with emphasis given to turning systems.
- Type of machining processes which can be performed by a particular turning systems. It should be noted that modern turning systems are able to perform, in addition to turning operation, processes such as milling, tapping and drilling.

The research work related to the definition and development of the Design for Injection Moulding application is being undertaken by R.J.V. Lee (1993). The structure and content for the Manufacturing Model in this particular work have been adopted from A. H. Al–Ashaab (1994).

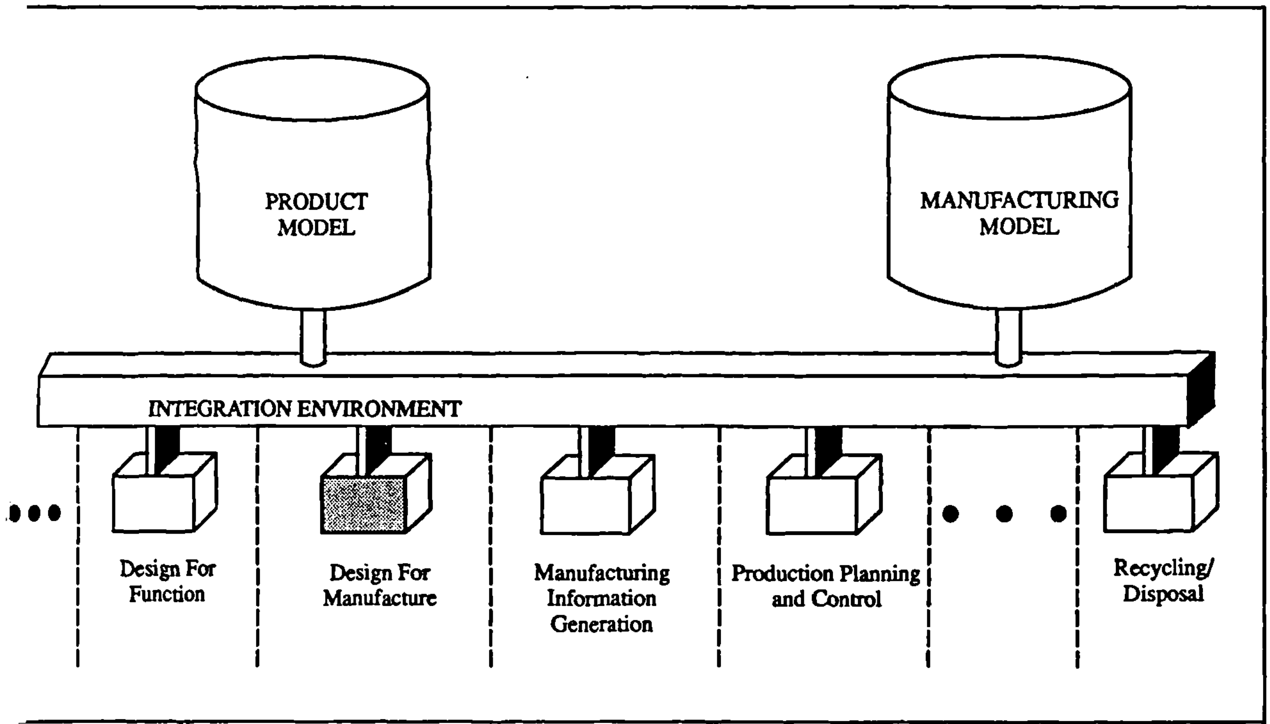
The scope of the Manufacturing Model has been extended in order to explore how manufacturing activities can be supported. Therefore a scenario related to the production planning of a product have been defined and are supported by the Manufacturing Model. This application is: NC machining planning. The research in the area of NC Machining Planning is based on the results of research carried out by Dr. A Tavakoli (Tavakoli 1993), a former PhD student in the laboratory.

5.5. The Wider Context for the Manufacturing Model

The Manufacturing Model describes and captures the information regarding the manufacturing facility of a particular enterprise in terms of its organization, composition and processes capabilities. Among all the different types of manufacturing facilities the author has focused in modelling the facilities used in batch manufacture. Nevertheless it is the author's opinion that it can be extended to cover other types of manufacturing organisations.

The Manufacturing Model represents the necessary information required to provide reliable manufacturing capability information for the support of life cycle activities, in particular the interaction between design for function and design for manufacture activities. It can also provide reliable information to support activities which generate information for manufacturing, such as process planning, machining planning, pre–processing proving, and scheduling. As the Manufacturing Model will represent the detailed manufacturing capability of an enterprise, and its current manufacturing status, this model will be an useful source of information for manufacturing activities such as Production Planning and Control (Figure 5.3). The extend of the applicability of the Manufacturing Model includes the design and manufacturing activities defined in the IDEF0 model described in the appendix B. Related to the manufacturing activities, the author has focused in exploring how the Manufacturing Model could support the activities related to plan how to manufacture and when to manufacture.

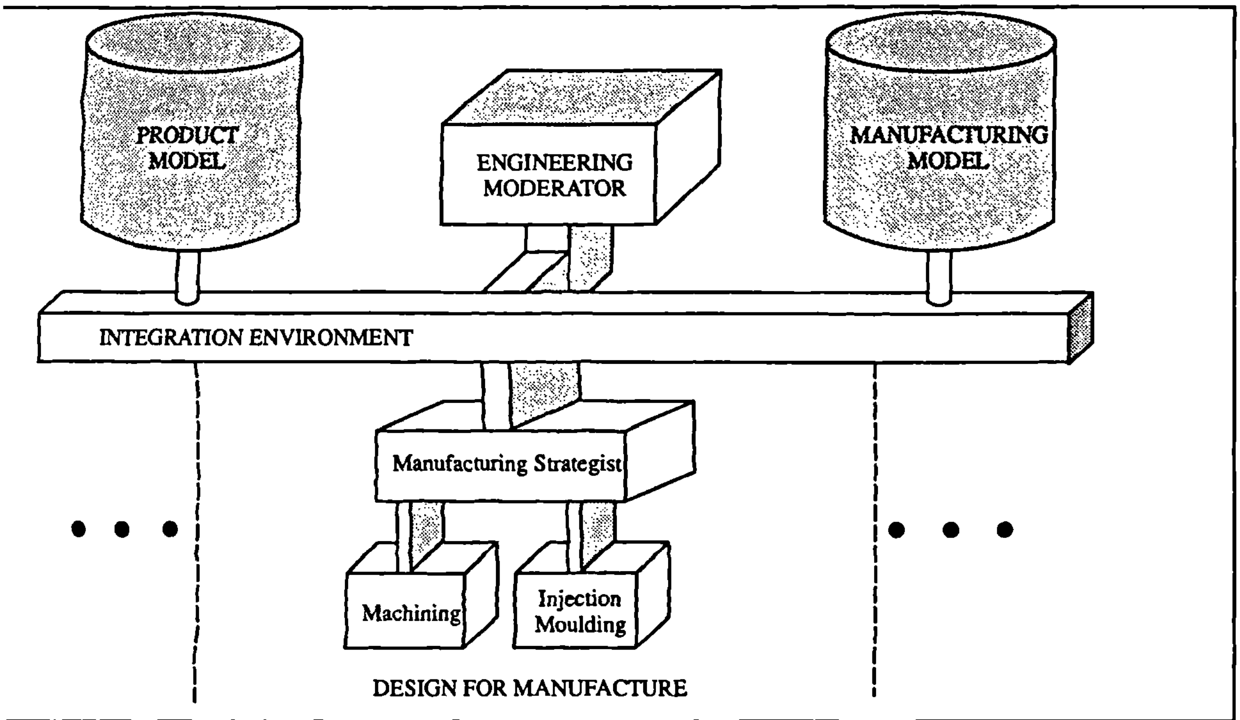
In real situations the necessary information to support design, manufacture and assembly activities will require in addition to the Product and Manufacturing Model, different databases such as material, procurement parts, and machining databases. In this way all the data necessary to design, manufacture and assemble a product will be readily accessible to the users and applications. The Product Model will hold information regarding the specification of a product, its components and manufacturing data to produce a specific product. Whilst the Manufacturing Model will provide the necessary information regarding the capabilities of resources and process available in a company to produce such a product. However information regarding available material to work with, standard parts needed during assembly or cutting data required to generate manufacturing information will have to be provided by others databases (Figure 5.4). These sort of databases will be incorporated to the CAE system and will be triggered by design, manufacturing or assembly applications when needed.



A. Molina

Figure 5.1 The MOSES Research Concept
(Adapted from ACME/SERC 1991 Proposal)

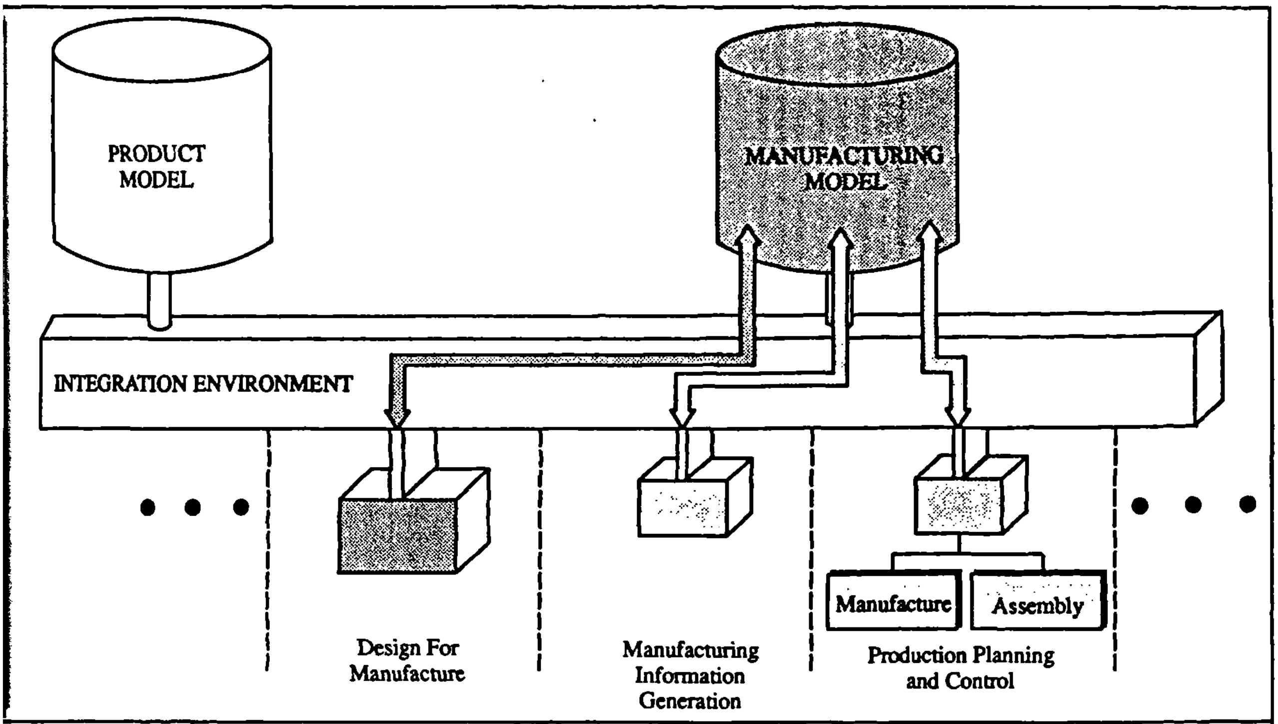
LUT–SE Research Group



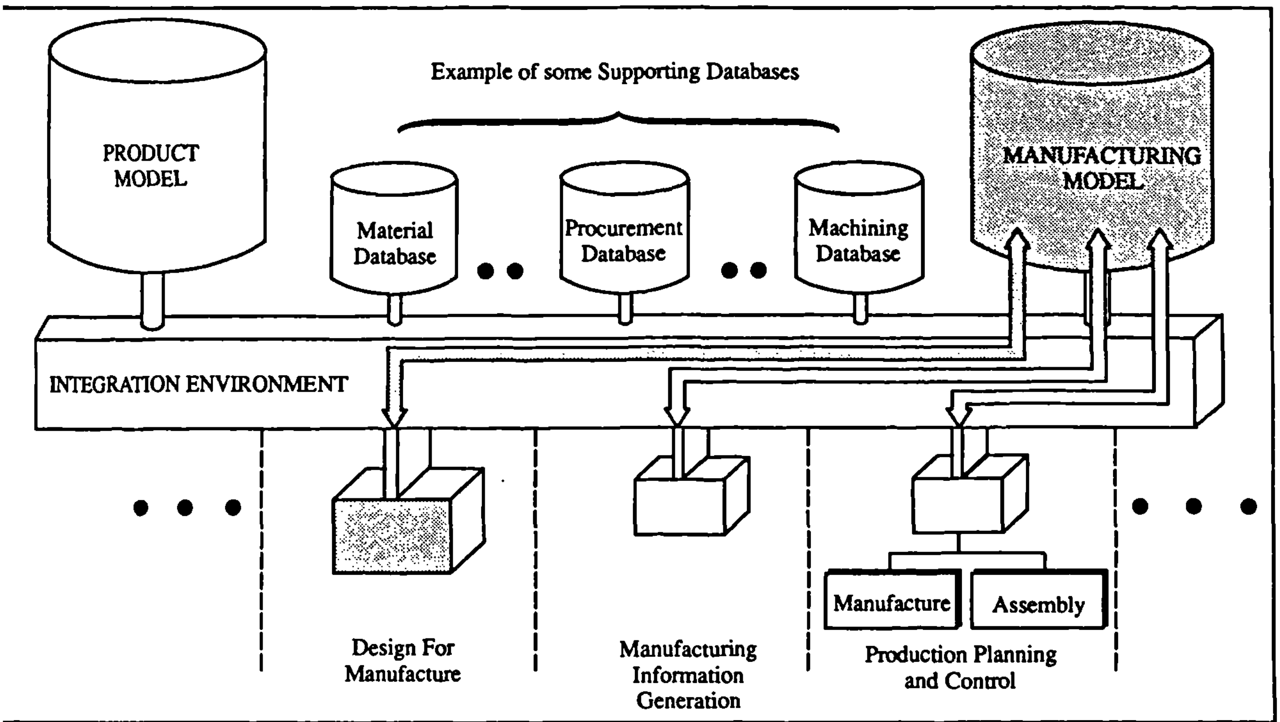
A. Molina

Figure 5.2 The Design for Manufacture Environment in the MOSES research project

LUT–SE Research Group



A. Molina	Figure 5.3 The Wider Context of the Manufacturing Model	LUT–SE Research Group



A. Molina	Figure 5.4 Product and Manufacturing Model supported by other databases	LUT–SE Research Group

Chapter 6

The Challenges and Decisions in Manufacturing Model Research

6.1. Introduction

This chapter describes the challenges faced by the author in the Manufacturing Model research. The author's decisions and contributions during the development of the Manufacturing Model are briefly presented. References are made to Chapter 5 and figures 5.1, 5.2 and 5.3 in order to set into a context the decisions outlined.

6.2. Definition of a Framework and Selection of Reference Models

A key issue in the MOSES research project (see Chapter 5, Section 5.1) is the development of a CAE Reference Model to provide a frame to support the development of existing and new CAE systems which support simultaneous engineering. The idea is to ensure that certain key issues are considered during the design of the CAE systems, and that standardised methods are used for the design and documentation of the systems. The author has contributed to the definition of the MOSES CAE Reference Model (CAE–RM) which is based on the Reference Model for Open Distributed Processing or RM–ODP (ISO/IEC 10746–1).

The research work was originally targeted on the use of the Reference Model for Open Distributed Processing (RM–ODP) to specify the MOSES CAE–RM. As the work evolved the need to use a complementary technique was identified by the author, in order to specify the enterprise requirements for CAE Systems based on the MOSES CAE–RM. The study of the literature led the author to analyse the possibilities of using reference models for integrating manufacturing enterprises (CIMOSA, Purdue Enterprise Reference Architecture and GRAI–GIM) to model such requirements. As a conclusion of this analysis, the CIMOSA reference model (ESPRIT Project 688/5288) was chosen to tackle this issue. The main reason for this choice was that CIMOSA provides a set of methods to systematically develop the requirements model needed in this research. The hybrid methodology introduces a formalism to supply the requirements level of the MOSES CAE–RM. This formalism is referred in this thesis as the CAE Framework.

The CAE Framework is shown in figure 6.1. The CIMOSA reference model is used to define the enterprise requirements for the use of a CAE system in a Concurrent Product Development Environment. The CIMOSA architecture is such that a CAE system, can be defined as a resource within the architecture by describing the required capabilities of the system. Once the required capabilities of the CAE system have been defined, the system can be designed and developed using the RM–ODP. The RM–ODP is used to guide the development of an open and flexible CAE System and its elements (e.g. Product Model, Manufacturing Model, etc). This reference model provides the underlying methodology required for the development of the Manufacturing Model. Although the RM–ODP allows the thorough description of an information system from different views, the research undertaken by the author was primarily involved with the definition of the first three viewpoints i.e. Enterprise, Information and Computation (Figure 6.2). The decision to only research on the first three viewpoints was based on the fact that these viewpoints are software independent and the main objective of this PhD research is to contribute with generic solutions to the problem of capturing, representing and providing manufacturing information for the support of concurrent product development (see Chapter 5 Section 5.4 for the definition of the scope of the Manufacturing Model).

The decision to make use of elements of these two Reference Models to form the CAE Framework required in this research was proved to be very significant. The use of the CIMOSA Requirements Definition Model in this hybrid methodology allows the assessment of the ways in which a particular RM–ODP system could provide support to a business. The extended use of this CAE Framework is discussed in detail in Chapter 7. The use of the CIMOSA Requirements Definition Model is presented in Chapter 8 and the use of RM–ODP as the basis for the MOSES CAE–RM is described in Chapter 9.

6.3. Selection of the Methodologies and Tools to satisfy the requirements of the Reference Model

The purpose of using the RM–ODP, as the basis for the MOSES CAE–RM, is to allow a model of a real system to be built, whether existing or planned. The enterprise may have requirements which are difficult to realise. Any gap between what can be achieved by the system and what is desired will be highlighted by a disparity in the mapping between the CIMOSA Enterprise Requirements Definition Model and the MOSES CAE–RM.

The RM–ODP allows the description of an information system from different viewpoints: Enterprise, Information, Computation, Engineering and Technology. Each view represents a specific aspect of the information system. The views of the RM–ODP allow a complex system to be described from a number of perspectives. The emphasis of each view is tailored to primarily represent either objectives, realisation or behavioural attributes. The RM–ODP defines a set of five languages, each corresponding to one of the viewpoints. Each language is defined to be used for the specification of an ODP system from the corresponding viewpoint. Therefore, it is important to select the modelling methodologies (or languages) that closely match the requirements imposed by each of the languages defined in the RM–ODP views.

From the methodologies described in Chapter 2 (Section 2.4), the following methodologies and languages were mapped into each of the relevant RM–ODP views of this research (Figure 6.3):

- Enterprise View: IDEF0
- Information View: IDEF0 and EXPRESS
- Computational View: Booch Method

In order to better support the use of these methodologies and languages, an evaluation of tools was carried out to select commercial tools for the Booch Object Oriented Methodology and EXPRESS language. No integrated tool was found, but the results of the study pointed out that for Booch the RATIONAL ROSE software was an adequate commercial tool available. On the other hand, for the development of STEP/EXPRESS models, the STEP Tool Kit (nowadays called ST–DEVELOPER) was chosen because of its advantages against other experimental and commercial tools. The implementation of a prototype version of the Manufacturing Model was carried out using the Object Oriented Database DEC Object/DB (Objectivity/DB 1991) and the Object Oriented Programming C++ (Stroustrup 1986), due to the fact, that these were the implementation tools provided by one of the industrial collaborators in the MOSES research project. The analysis and selection process of the Reference Model Methodologies and Tools used in this research are described in detail in Chapter 10.

6.4. The creation of a Manufacturing Information Modelling Methodology as a formalism for the development of the Manufacturing Model

During the development of this research and based on the literature reviewed, specially object oriented modelling techniques (Booch 1991, Wirfs–Brock et al. 1990, Coad and Yourdon 1990, Rumbaugh et al. 1991), the author recognised that different modelling methodologies were required in order to adequately describe a manufacturing facility and its capabilities. The need to have an underlaying methodology which could be used as the foundations for underpinning the Manufacturing Model research lead the author to conceive a novel Manufacturing Information Modelling Methodology. These methodology embraces concepts from different modelling techniques. The concepts introduced in this methodology are presented in detail in Chapter 11, and its use is described in Chapter 12. However in the next subsections key issues of this methodology are highlighted to allow the reader to appreciate their importance.

6.4.1. Specification of the aspects to model about a Manufacturing Facility

The literature reviewed in hybrid methodologies (Chapter 2, Subsection 2.4.4), together with past research experience (Molina et al. 1992), have lead the author to recognise that different modelling dimensions are required in order to adequately describe a manufacturing facility from different perspectives. These modelling dimensions are related to the description of the data, function and behaviour of the entities that exist in the universe of discourse, in this case: the manufacturing facility. Thus the modelling of the data, function and behaviour of manufacturing facilities and their elements has been identified as a key issue in the development of a sound and consistent Manufacturing Model.

Being able to model all these three related dimensions in the Manufacturing Model allows a suitable description of the manufacturing facility in terms of its composition, organization, and capabilities. Details about these modelling dimensions are presented in Chapter 11, Section 11.3, especially, the modelling of data and function have been fully developed. Nevertheless, it is important to mention that the behaviour modelling dimension was not fully explored in this research. Due to the fact that the scope of the Manufacturing Model was limited to support concurrent design of products i.e. to support the Manufacturing Strategist and the Design for Machining applications of the MOSES research project (Chapter 5, Section 5.3 – 5.4, Figure 5.2). The type of information needed for these applications required only the data and function to be modelled, thus the need to include the behaviour dimension in the Manufacturing Model was not necessary during this PhD research. However the author considered the inclusion of the behaviour dimension to be important, even when it was not developed, in the manufacturing information modelling methodology described in Chapter 11, Section 11.2 as it is necessary to define a complete and coherent frame of work.

6.4.2. Identification of the Manufacturing Information Entities

The results of research undertaken by IMPPACT (IMPPACT 1991) and CIMOSA (ESPRIT Project 668/5288) have shown that at least two information entities are required to define a manufacturing environment i.e. manufacturing resources and processes. The representation of structured resources and processes allows an ad–hoc representation of the manufacturing facilities and capabilities in terms of process technology and equipment. But, in the author's opinion, in addition to this type of information, there is a need to represent the company's strategic decisions and operational rules, here called manufacturing strategies, because these manufacturing strategies describe constraints imposed on the use and the organization of resources and processes. This new aspect captures the company's manufacturing strategy. Therefore, the author has defined three information entities as core elements in the definition of the Manufacturing Model: resources, processes and strategies (Figure 6.4). The relations and interaction amongst them describes the manufacturing capability of a company. This is required in order to be able to support the manufacturing information needs of the Application Environments which assist the realization of the activities throughout the product's life cycle (Chapter 5, Section 5.1, Figure 5.1).

The manufacturing resources are all the physical elements within a facility which enable product realization such as: production machinery, production tools, material handling equipment, storage systems, humans, supply and disposal units, etc. The resources can be organized in groups to create manufacturing facilities such as stations, cells or shop–floors. Early in this research, the author recognised the need to represent the resources in a function oriented manner in order to describe their role in supporting the design and manufacturing activities. Thus, the description of the resources are based on their physical properties and functional composition to allow the characterisation of their capabilities. The representation of resources in the Manufacturing Model is similar to the ones defined in IMPPACT and KCIM. In addition, whenever possible British Standards were employed (Molina 1993). The detailed description of the research carried out by the author in modelling the manufacturing resources is presented in Chapter 12, Section 12.4.

The manufacturing processes used in a facility and performed by resources can be classified into two types: information and material processes. In this research the representation of the capabilities of material processes, such as turning, drilling, milling and assembly processes is of major importance. Different generic representations were analysed and evaluated in order to determine the best way to describe a manufacturing process e.g IDEF0, Enterprise Activity (CIMOSA) or Generic Activity Model (BSI DD 203: Part1 : 1991). A generic definition of the process based on the Generic Activity

Model, which can easily be mapped into the Enterprise Activity construct of CIMOSA, has been defined in the Manufacturing Model. Using this description a detailed representation of most manufacturing processes can be made. The author has focused on representing the capabilities of the machining process. These capabilities are constrained by the capabilities of the resources which can perform the machining process. Therefore, the author decided to represent these capabilities by using relations between the possible movements between the machine components (spindle, turret, work table, etc.) and the production tools (tool holders and cutters). Chapter 12, Section 12.5 describes the models employed by the author to represent the manufacturing process capabilities.

Two type of decisions were identified by the author, which made possible the formulation of manufacturing strategies: decisions made over time which define the organisation, composition and capacity of the manufacturing facilities, and the day to day decisions which determine how to use the facilities and related capabilities. In the Manufacturing Model, strategies represent how the resources and processes are organized, composed and used to support the realization of the manufacturing function in order to achieve the manufacturing objectives of a company. Relevant to this research on modelling strategies, from the eight classes suggested by Hayes and Wheelwrigh (1984), the author has considered only strategies related to capacity, facility structure, technology and production planning/material control. In Chapter 12, Section 12.6 details are explained about how the manufacturing strategies were modelled.

6.4.3. Definition of the structure to represent the Manufacturing Information Entities

One of the major challenges in this PhD research is the representation of the capability of a manufacturing facility from different perspectives. The way a manufacturing engineer sees the capability of a facility differs from the one of a shop–floor manager. Thus it is important to be able to represent the facility from different points of view in order to provide manufacturing information to a diversity of users e.g. design engineers to do design for manufacture, manufacturing engineers to produce process plans and NC–code and shop–floor managers to formulate and execute schedules (Figure 6.5). The author considers that in order to support an information structure adequate to represent the capability of a manufacturing facility, the Manufacturing Model should:

- be generic in order to support general and specific representation of different enterprises,
- be able to capture the manufacturing information at different levels of functionality i.e. from different perspectives,
- be composed and crganized in a manner which facilitates the communication of manufacturing information

A generic representation of the manufacturing facility is achievable by defining the manufacturing resources, process and strategies based on their generic attributes and classifying them in terms of these characteristics. It is important to allow the generic definition of resources and processes, and the particular description of strategies in order to support general and enterprise specific characterizations. Major challenges that are faced in the representation of generic structures are:

- the need to use high level of abstractions to represent generic concepts
- the non existence of standard classification (taxonomies) of resources, processes or strategies
- the wide range of attributes that manufacturing entities have to include in their definition

In order to cope with these challenges, a four level Manufacturing Model has been defined (Figure 6.6). The idea of using a four level model is to comply with the different levels of functionality common to manufacturing enterprises in order to capture the manufacturing function. In addition, taxonomies were developed to represent hierarchical groups of resources, processes and strategies, and only the necessary attributes required for each information entity were considered.

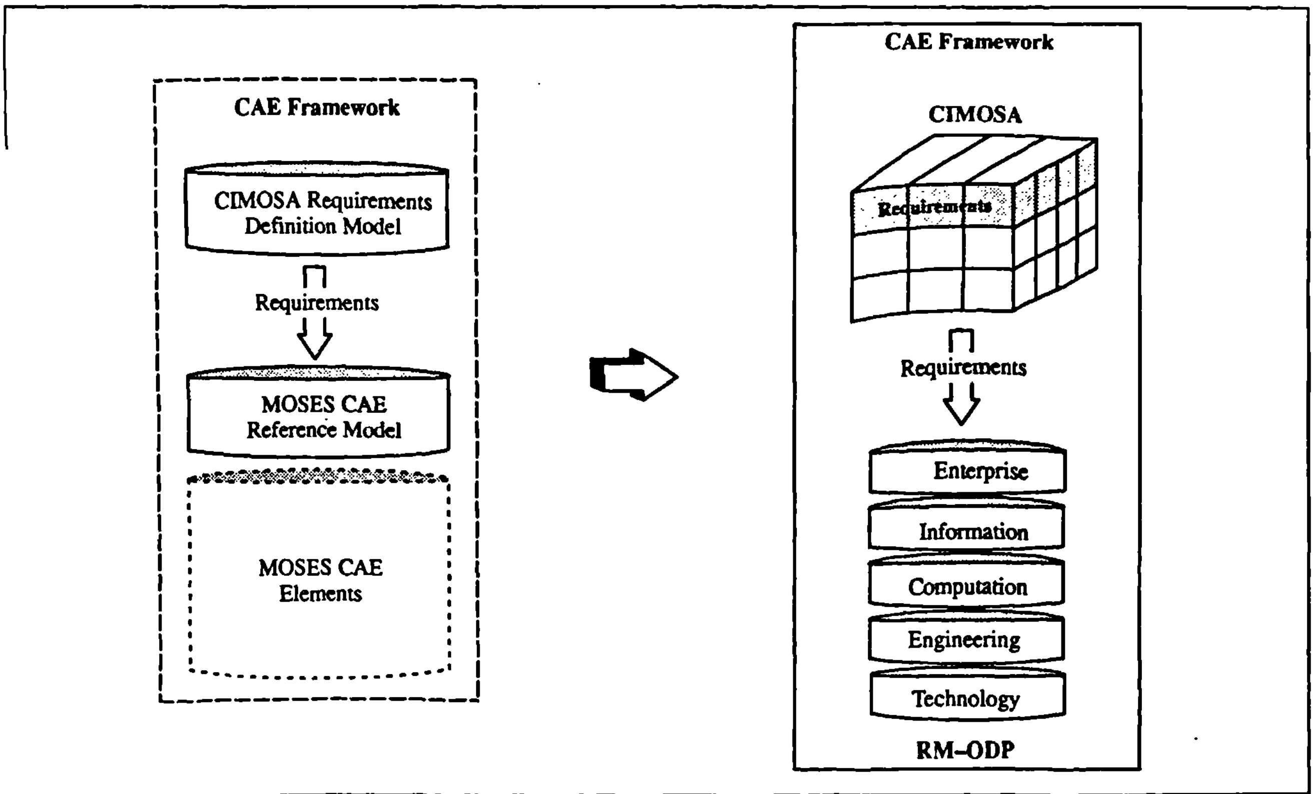
Nowadays, manufacturing enterprises in general have adopted five or four functional levels to realize their function. A de–facto standard has already emerged, by the petition of various standardization bodies such as ISO and NIST (National Institute of Standards and Technology), together with various European projects within ESPRIT (BSI PD 6526:1990). This de–facto standard had partitioned the factory into five hierarchical levels: Facility (Factory), Shop (Area), Cell, Work station and Equipment.

This five hierarchical model has been used as a reference to structure our four level Manufacturing Model with the following levels (Figure 6.7):

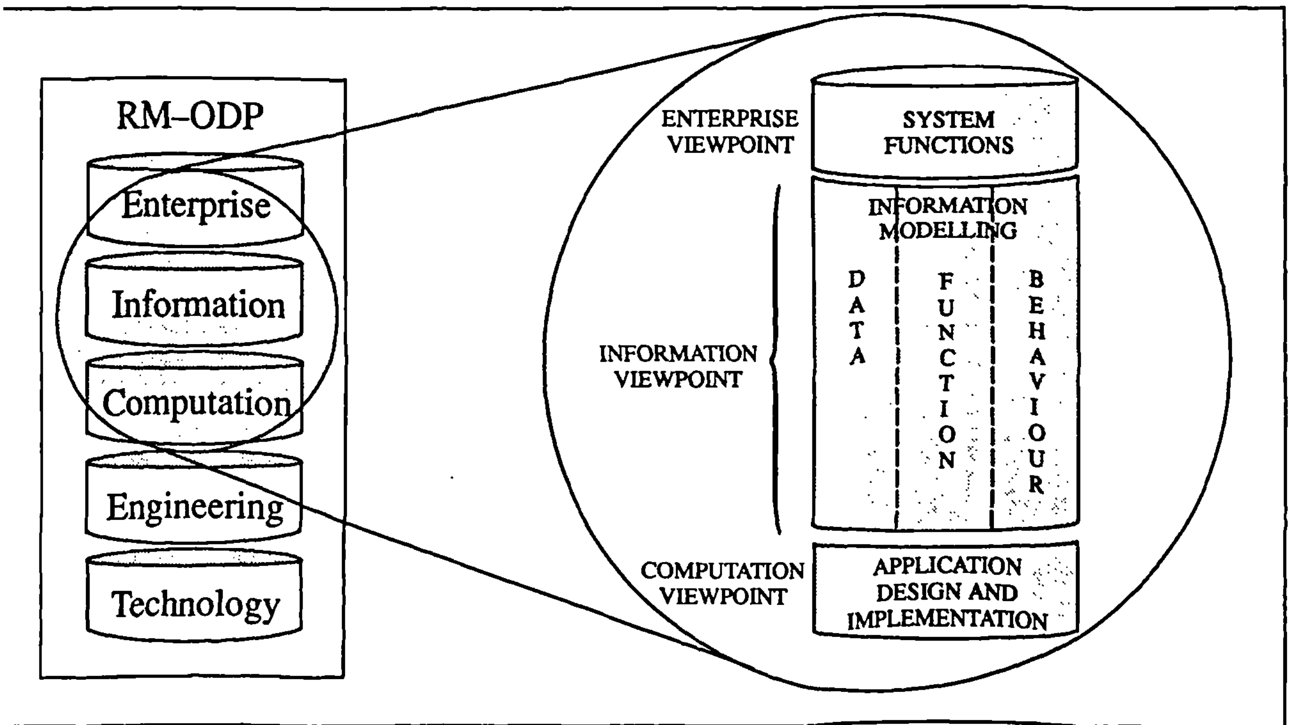
1. Factory Level
2. Shop Level
3. Cell Level
4. Station Level

where the fifth level (Equipment) is enclosed at the Station Level. These levels of abstraction provide manufacturing information for all functional hierarchical activities within a manufacturing enterprise. This hierarchical organisation will enable the Manufacturing Model to provide the adequate manufacturing information to the different application environments described in Chapter 5, Section 5.5 (see Figure 5.3) i.e. Design for Manufacture, Manufacturing Information Generation, and Production Planning and Control.

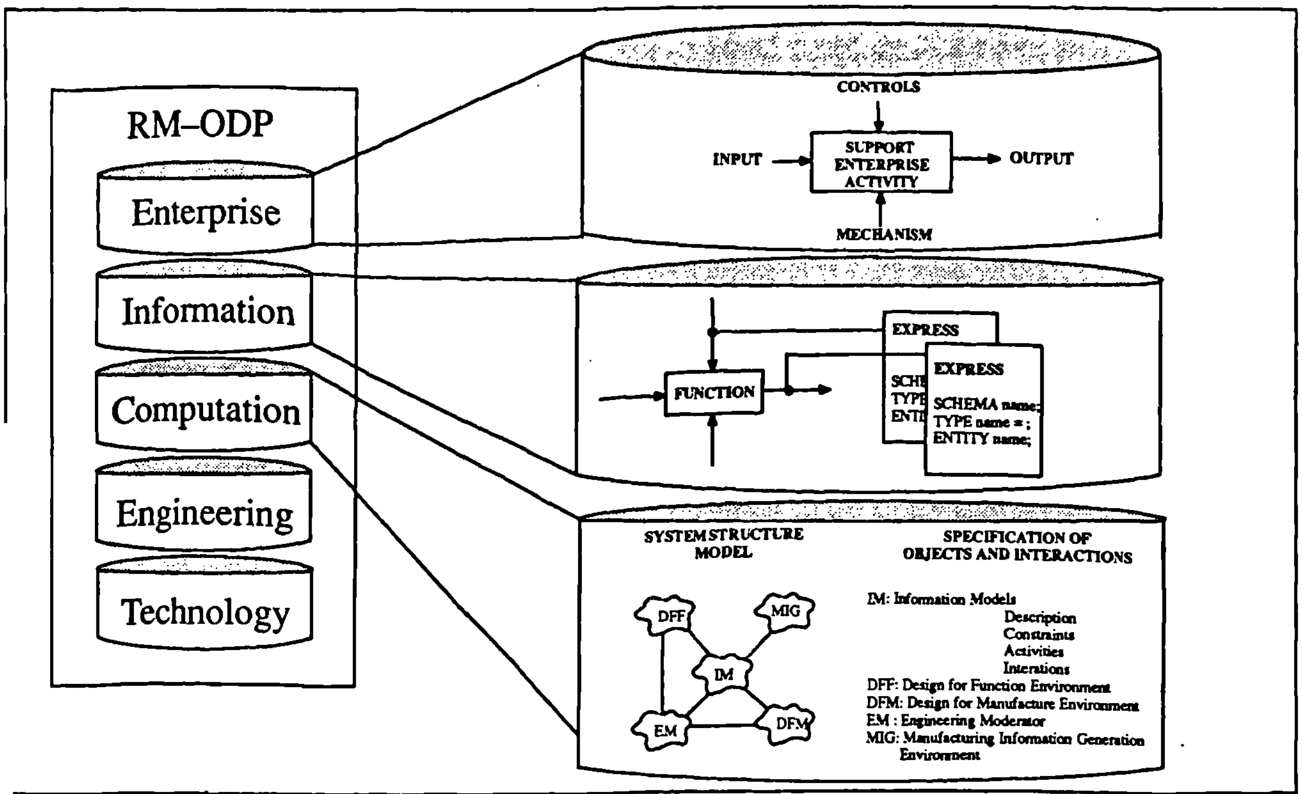
The necessary composition and organization of the information entities (i.e. resources, processes and strategies) in this hierarchical Manufacturing Model, together with the taxonomies defined for each of the information entities constitutes a major component of the research reported in this thesis. A detailed discussion of each of these levels and their functionality is presented in Chapter 11, Section 11.5.



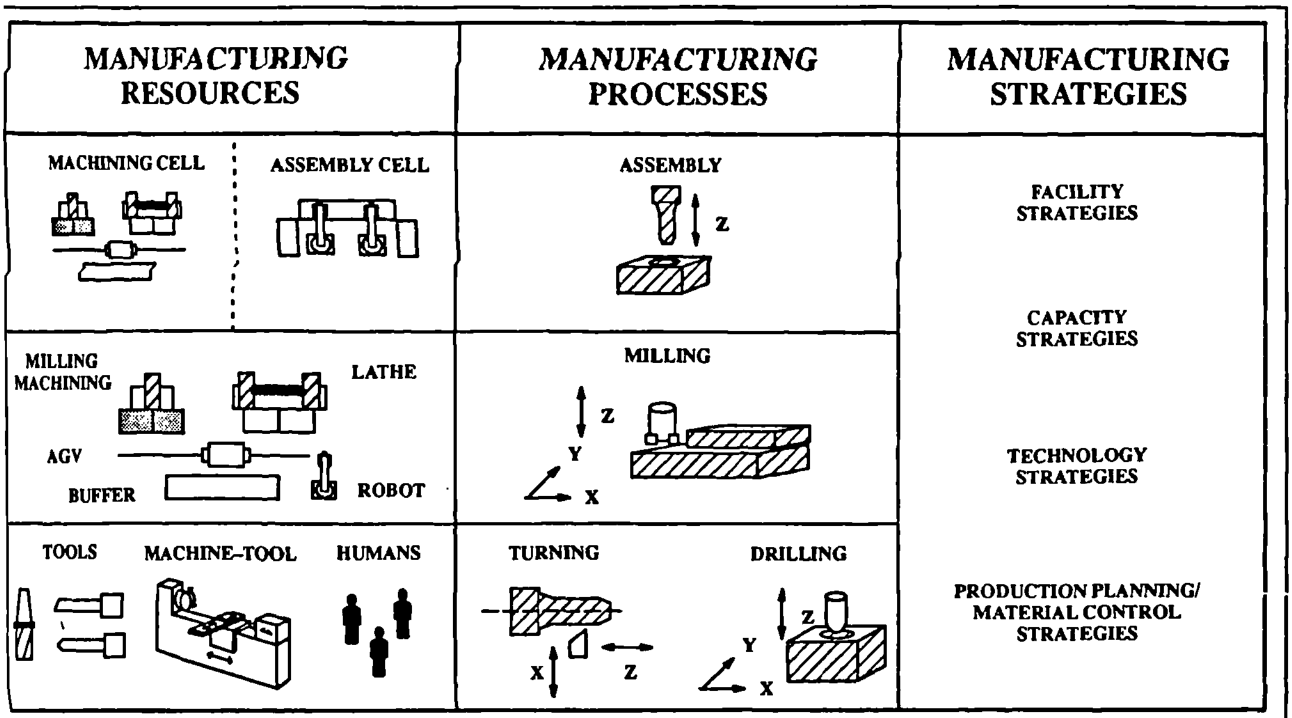
A. Molina	Figure 6.1 CAE Framework based on CIMOSA Requirements Definition Model and MOSES CAE–RM (RM–ODP)	LUT–SE Research Group



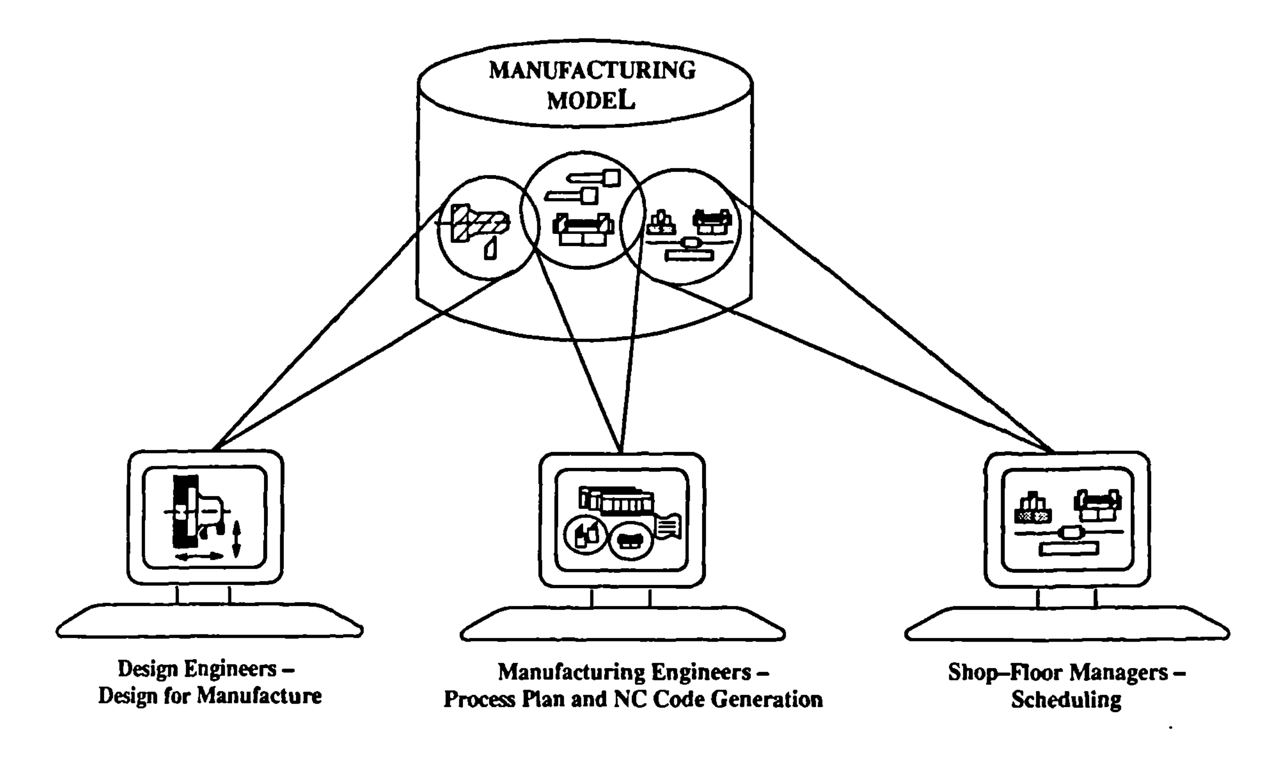
A. Molina	Figure 6.2 The Reference Model of Open Distributed Processing (RM–ODP) to guide the development of the CAE System and Manufacturing Model	LUT–SE Research Group



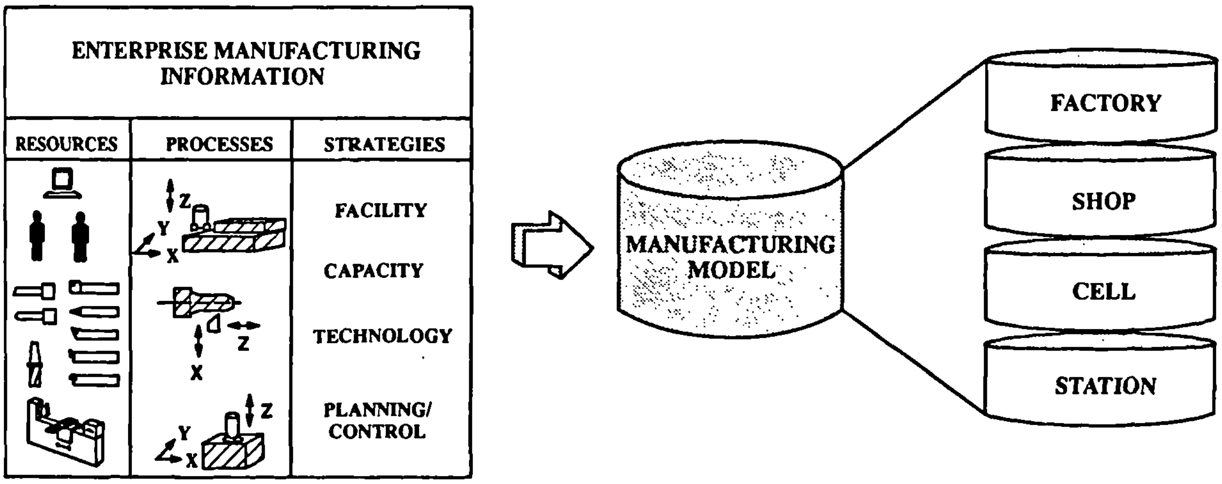
A. Molina	Figure 6.3 Mapped Methodologies to the RM–ODP	LUT–SE Research Group



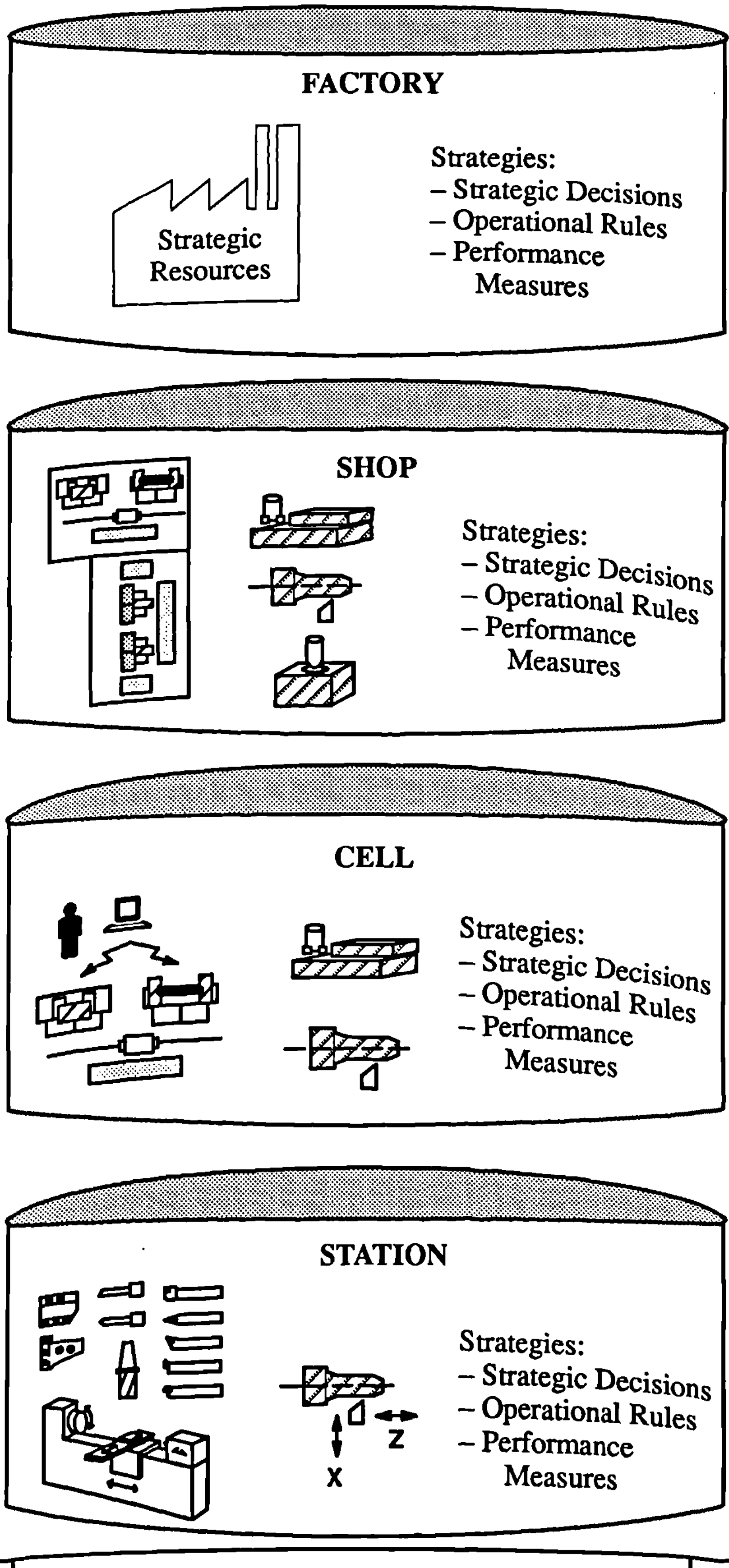
A. Molina	Figure 6.4 Manufacturing Information Entities	LUT–SE Research Group



A. Molina	Figure 6.5 Different perspectives of the Manufacturing Model	LUT-SE Research Group



A. Molina	Figure 6.6 Manufacturing Information Captured in a Four Level Manufacturing Model	LUT-SE Research Group



A. Molina	Figure 6.7 Manufacturing Model Structure and Content	LUT–SE Research Group

Chapter 7

The Hybrid Methodology used to define the CAE Framework of this Research

7.1. Introduction

This chapter explains and describes the decisions made on the definition of the formalism used in this research named the CAE Framework. The arguments for the selection of the Open System Architecture for Computer Integrated Manufacturing (CIMOSA) and the Reference Model for Open Distributed Processing (RM–ODP) to specify the CAE Framework are discussed. The author's contributions to the definition of the MOSES CAE Reference Model (CAE–RM) based on the RM–ODP are described. The relation between the CIMOSA Requirements Definition Model and the Enterprise Viewpoint of the MOSES CAE Reference Model is outlined.

7.2. The Definition of the CAE Framework

At the beginning of the research the use of the Reference Model for Open Distributed Processing (RM–ODP) to define the MOSES CAE Reference Model (CAE–RM) was specified (Chapter 5, Section 5.1). Nevertheless, the author found, as the work evolved, the need to use a complementary methodology in order to specify the enterprise requirements for CAE Systems. The idea was to define a context where the CAE systems developed based on the MOSES CAE–RM could be used and integrated. The result of this search for a formalism was the hybrid methodology based on the combined use of CIMOSA Requirements Definition Model and the MOSES CAE Reference Model. This formalism is the CAE Framework.

The rationale of the CAE Framework is based on the idea that a formal definition of enterprise requirements can be used as the driver for the selection of alternative CAE systems. The match between the particular needs of an enterprise against different available CAE systems can be facilitated, if the CAE system concepts are represented in a reference model which allows the description of the CAE system functionality and elements (Figure 7.1). This concept has been introduced in projects such as IMPPACT (Gielingh and Suhm 1994) and CIMOSA (ESPRIT Project 688/5288). In IMPPACT, the IMPPACT CIM Reference Model describes a generic specification of conceptual models for information integration between CIM components and their functional behaviour. In the CIMOSA architectural framework, a Design Specification Model is used to describe and evaluate alternative technical designs in order to select the best available technical solution. A similar concept has been defined in the MOSES research project to represent and develop the CAE system concepts described in Chapter 5 Section 5.1. The MOSES CAE Reference Model specifies the functionality of the MOSES CAE system and how this functionality is implemented by using a multi–level representation based on RM–ODP.

In order to formalize the definition of enterprise requirements, frameworks have been defined in IMPPACT and CIMOSA which include the model needed to perform the elicitation of those requirements (Figure 7.2). The IMPPACT Business Reference Model defines an enterprise in terms of the functions and information flows, which allow, among other things, the identification of the needs for new information systems. The CIMOSA Requirements Definition Model provides the means to gather and express the business requirements of a particular enterprise from different views i.e. function, information, resources and organisation.

This idea of using a model to define the enterprise requirements has been introduced by the author in this research in order to define the CAE framework (Figure 7.2). The use of a requirements model allows different enterprises to specify what functionality is required for a CAE system to support design or manufacturing activities. In order to define the enterprise requirements in a formal manner, the author decided to use the Requirements Definition Model of the CIMOSA architectural framework.. The main reason for this choice was that CIMOSA provides a set of methods to systematically develop the requirements model needed in this research.

The CAE Framework is therefore composed of two reference models (Figure 7.3):

1. The CIMOSA Requirements Definition Model
2. The MOSES CAE Reference Model based on the Reference Model for Open Distributed Processing (RM–ODP)

The author has introduced the use of the CIMOSA Requirements Definition Model to represent the enterprise requirements. This model allows the description of those requirements using four views: function, information, resources and organisation. In this research this model is used to specify the CAE system requirements (Figure 7.3).

In the MOSES research project the use of the Reference Model for Open Distributed Processing (RM–ODP) to define the MOSES CAE Reference Model (CAE–RM) was specified in order to have a multi–viewpoint representation of the MOSES CAE system concepts (Chapter 5 Section 5.1). The viewpoints of the CAE–RM are: Enterprise, Information, Computation, Engineering, Technology. The CAE–RM describes the CAE system functionality, configuration and technology that is necessary to satisfy the requirements specified by the CIMOSA Requirements Definition Model. An schematic functionality of the system is illustrated in the figure 7.3.

The CAE Framework aims to assist the system users and developers in the following tasks:

- Identify the enterprise's CAE system requirements to support simultaneous engineering i.e. functionality, information, resources, role in the organization, etc.
- Guide the design and implementation of the CAE system itself.
- Organize the models, methods and tools which allows the evolution of the CAE system towards the desired level of integration and automation.

The methodological approach employed by the author in the building of the CAE Framework is summarized below to help the reader appreciate the sections that follow:

1. The CIMOSA architecture framework was studied to understand the underlying concepts and set of methods employed to build an enterprise model.

2. A Partial Enterprise Requirements Definition Model was built using the CIMOSA modelling methodology in order to represent an enterprise which performs product development by using simultaneous engineering principles.

3. The Partial Enterprise Requirements Definition Model was instantiated to represent the CAE system requirements of a particular enterprise.

4. The RM–ODP was used to guide the design and development of the MOSES CAE System according to the requirements defined in the CIMOSA Requirements Definition Model for that particular enterprise. The set of partial models and methods employed to represent the functionality, configuration and technology of the MOSES CAE System constitute the MOSES CAE Reference Model (CAE–RM).

The CAE Framework is therefore an hybrid methodology based on the CIMOSA Requirements Definitions Model and the multi–viewpoint MOSES CAE–RM based on RM–ODP. The CAE Framework provides successful support for the identification of requirements of a CAE system, the provision of guide-lines for CAE system development and a frame for the organisation of models, methods and tools to describe, represent and build CAE systems.

This CAE Framework allows the assessment of the ways in which a particular CAE system, developed using the CAE–RM, could provide support to different CIMOSA enterprise requirements models. In this thesis, in order to demonstrate this concept, a Partial CIMOSA Enterprise Requirements Definition Model was developed and instantiated, and the functionality of the MOSES CAE–RM was matched to this model in order to establish the important link between requirements to be satisfied and the system's functions which satisfy those requirements (see Chapter 8 for the description of this model, its instantiation and an example of the match between requirements and CAE system functions).

7.3. Describing CAE System Requirements using the CIMOSA Requirements Definition Model

During the development of this research project, the author recognised that an important task for the successful computer support of simultaneous engineering is the identification of the enterprise requirements i.e. where the Computer Aided Simultaneous Engineering System (CAE system) is going to be installed, what its functionality has to be and how it is operated. Being able to recognise these needs enables the CAE system to be configured according to that particular manufacturing environment. Based on the literature on enterprise modelling (Chapter 2, Section 2.5), the author identified different aspects which have to be considered to capture enterprise requirements, among others: the manufacturing strategy, the activities that require support, flow of information, availability of resources, organization and responsibilities, etc. A key problem is to determine the method or set of methods to be used in order to capture all these aspects in a formal representation, and hence guide the development of an integrated information system. Experiences on research in the area of enterprise modelling (Williams et al. 1993) have demonstrated that all these aspects about an enterprise can be captured in an Enterprise Model. Enterprise Models can be developed using reference models such as CIMOSA (ESPRIT Project 688/5288), GRAI–GIM (Doumeingts et al. 1992) and the Purdue Enterprise Reference Architecture (Williams 1991). From these reference models the author has chosen CIMOSA to represent the enterprise requirements to develop products using simultaneous engineering principles, and therefore, the required capabilities for a CAE System to support simultaneous engineering. CIMOSA was selected due to its formal approach employed in system modelling.

The CIMOSA Reference Model allows the construction of enterprise models. In order to build this model the author followed the CIMOSA modelling process which requires the enterprise model to be built in terms of four interrelated views: function, information, resources and organisation (Figure 7.4). By modelling the function view, the activities to design, manufacture and maintain products with consideration for the whole product life-cycle are identified. The model developed in this research only covers the activities related to the concurrent design of products, and not all the activities in the enterprise. During the modelling of these activities the information required to realize these activities is determined. This information is modelled and captured in the information view. By combining the results of the information and function views the required capabilities for the CAE system to support simultaneous engineering, as a resource, can be defined. Thus, the CAE system capabilities form the resource view. The organisation models the responsibilities and authorities of the group of people who have influence over the elements defined in the function view, information view and resource view (in this case the Simultaneous Engineering Team). This team is responsible for the realization of the concurrent design of products.

Figure 7.4 describes, in general, the phases of the CIMOSA modelling process: Domain Establishment, Decomposition of Domain into Business Processes and Enterprise Activities and finally, definition of Enterprise Activities, Information and the Organisation. Once the CIMOSA Requirements Definition Model was defined, this model was instantiated to specify the CAE system requirements for a particular enterprise. The CAE system requirements are defined in terms of:

- the Enterprise Activities to be supported (function view)
- the information which is required in those Enterprise Activities (information view)
- the elements of the CAE system which are needed to support those Enterprise Activities (resources view)

- the role of people involve in the specification of those requirements (organisation view)

The CIMOSA model is therefore a driver to define the functions required by a CAE system. The author believes that the use of CIMOSA allows the creation of a more structured and flexible CAE Framework. These characteristics enable the framework to be based upon an available, acceptable and formal terminology and methodology; and perhaps a future standard. In addition, the model can be applied to a wide range of systems within the domain of information systems to support enterprise integration.

The author's major reasons for chosing the CIMOSA reference model to be part of the CAE Framework were:

- it is a well defined and documented reference model
- it has a formal approach to system modelling
- the CIMOSA concepts match the ones required for the definition of the CAE–RM at the enterprise view
- it has the potential of becoming an international standard

It is the author's opinion that the use of CIMOSA in the CAE Framework allows future CAE systems, developed based on the MOSES CAE–RM, to be easily integrated within the enterprise. This integration could be easily achieved because the system requirements are clearly defined and may be used as the drivers for the development of the CAE system.

7.4. The Definition of MOSES CAE Reference Model based on RM–ODP

Once the enterprise requirements for the computer support of simultaneous engineering have been identified and defined using CIMOSA, the author faced the challenge to provide a description of a CAE system and its functionality to match and satisfy those requirements. The author uses the MOSES CAE Reference Model (CAE–RM) for this purpose and in particular the Enterprise Viewpoint.

The purpose of using the CAE–RM is to model a real system, whether existing or planned. The enterprise may have requirements which are difficult to realise. Any gap between what can be achieved by the system and what is desired will be highlighted by a disparity in the mapping between the CIMOSA Enterprise Requirements Definition Model and the Enterprise viewpoint of the CAE–RM.

On this basis, the design, configuration and implementation of the CAE system can be undertaken by selecting the system elements, defined in the CAE–RM, which are important for that particular manufacturing environment and satisfy the enterprise requirements. The system elements could be: information models (e.g. Product and Manufacturing Models), decision support environments (e.g. Design for Function, Design for Manufacture, etc.), and the adequate integration infrastructure (i.e. information system architecture).

The MOSES CAE–RM is based on the RM–ODP. The RM–ODP is a five level model which is intended to represent open distributed systems. To achieve this, the following five levels have been defined: Enterprise, Information, Computation, Engineering, and Technology (Figure 7.5). The following text briefly describes the different viewpoints of the MOSES CAE–RM to set the context for the discussion of the next section. The reader can find a detailed specification of the MOSES CAE–RM in Chapter 9, Section 9.2.

The Enterprise Viewpoint is associated with the specification of the CAE system functions. The Information Viewpoint focuses on describing the semantics of information and information processing functions in the CAE system. The Computational Viewpoint establishes the representation of the functional decomposition of the CAE system into elements. The Engineering Viewpoint focuses on the infrastructure required to support distribution. Finally, the Technological Viewpoint focuses on the selection of the necessary technology to support the CAE system.

7.5. Describing CAE System Functionality using the Enterprise Viewpoint of the MOSES CAE Reference Model

One of the most important decisions made by the author during this research was related to the definition of the scope of the information and the level of detail contained in the Enterprise Viewpoint. Two arguments were debated and are still being discussed in the MOSES Research project. The two arguments are:

1. The Enterprise viewpoint should describe the organisational environment, in terms of engineering functions and the relationships between them, in which CAE systems could be used.

2. The Enterprise viewpoint should represent the functional capability of CAE system.

In order to be able to make a decision in this PhD research, according to what the standard specifies, the author reviewed in detail the document ISO/IEC 10746–1. In page 66 of that document the follc wing text reads : "An ODP specification should not be confused with a total enterprise specification: the ODP enterprise specification is limited to the description of ODP functions". Based on this findings, the author has agreed, with other members of the MOSES LUT research project, to represent at the Enterprise Viewpoint the functionality that the CAE system is intended to achieve in order to support the different enterprise activities described in the CIMOSA Requirements Definition Model. This argument enables the Enterprise View to be the link between the two reference models i.e. CIMOSA and MOSES CAE–RM. The concept behind this is to design CAE Systems based on the MOSES CAE–RM which could match the requirements of CIMOSA Requirements Definition Models.

The Enterprise Viewpoint of the MOSES CAE–RM, therefore represents the functions provided by the CAE system and its elements (e.g. Manufacturing Model, Design for Manufacture Environment, Engineering Moderator) in order to satisfy the requirements of CIMOSA Requirements Definition Models.

Figure 7.5 describes what the author has defined as the functionality of the CAE system. This schematic representation of the functionality represents a system which provides both support for different enterprise activities (i.e. activity 1 ... activity n) and reliable information to perform such activities. In practice, the enterprise activities defined by the CIMOSA Requirements Definition Model which are carried out by a simultaneous engineering team (SE Team) are supported by the CAE system. The SE team is assisted to achieve the concurrent product development by inquiring the CAE system and receiving from the system a response. The response can be to advise regarding a design choice, information regarding a product, information regarding a machine capability, etc. From the users point of view, there are two important elements in the CAE system:

1. elements which support the life cycle activities i.e. the application environments (e.g. Design for Function Environment, Design for Manufacture Environment, etc.)

2. elements which provide reliable information for the product life cycle i.e. Product and Manufacturing Models.

Between these two type of elements there are interactions (input/response) which allows the CAE system to provide the functions required by the user. Nevertheless, these interaction are transparent for the user, and the system is seen as an integrated application which support the SE team in the realization of the enterprise activities required for concurrent product development.

In the MOSES CAE–RM, the Enterprise Viewpoint is described using IDEF0 activity diagrams (Figure 7.5). The IDEF0 methodology was used as the formal method to specify the CAE system functions. The detailed explanation of why IDEF0 has been chosen to represent this viewpoint is in Chapter 10, Subsection 10.3.1. It was very important to define the IDEF0 model in a way that the link between the CIMOSA requirements could be matched to the functions provided by the CAE system and its elements. In the IDEF0 method this has to be done by specifying at the top level diagram what the system does. Two possibilities arose to define this:

1. Represent the functionality as : "Support Simultaneous Engineering Team"

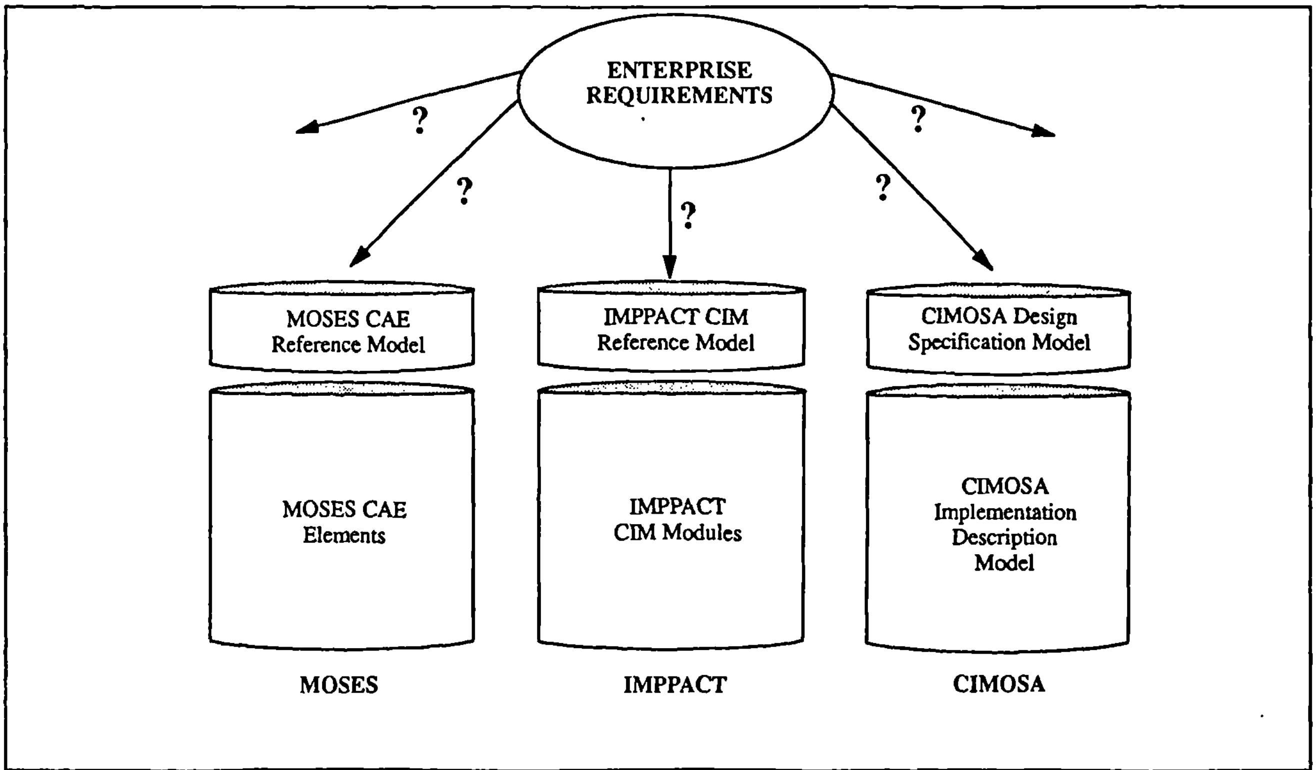
2. Represent the functionality as: "Support Enterprise Activity".

It was agreed to define the top level diagram as the generic function: "Support Enterprise Activity". In this manner all the functions provided by the CAE system elements support enterprise activities, and by doing so, each IDEF0 activity representing a function of the CAE system could be linked directly to an enterprise activity.

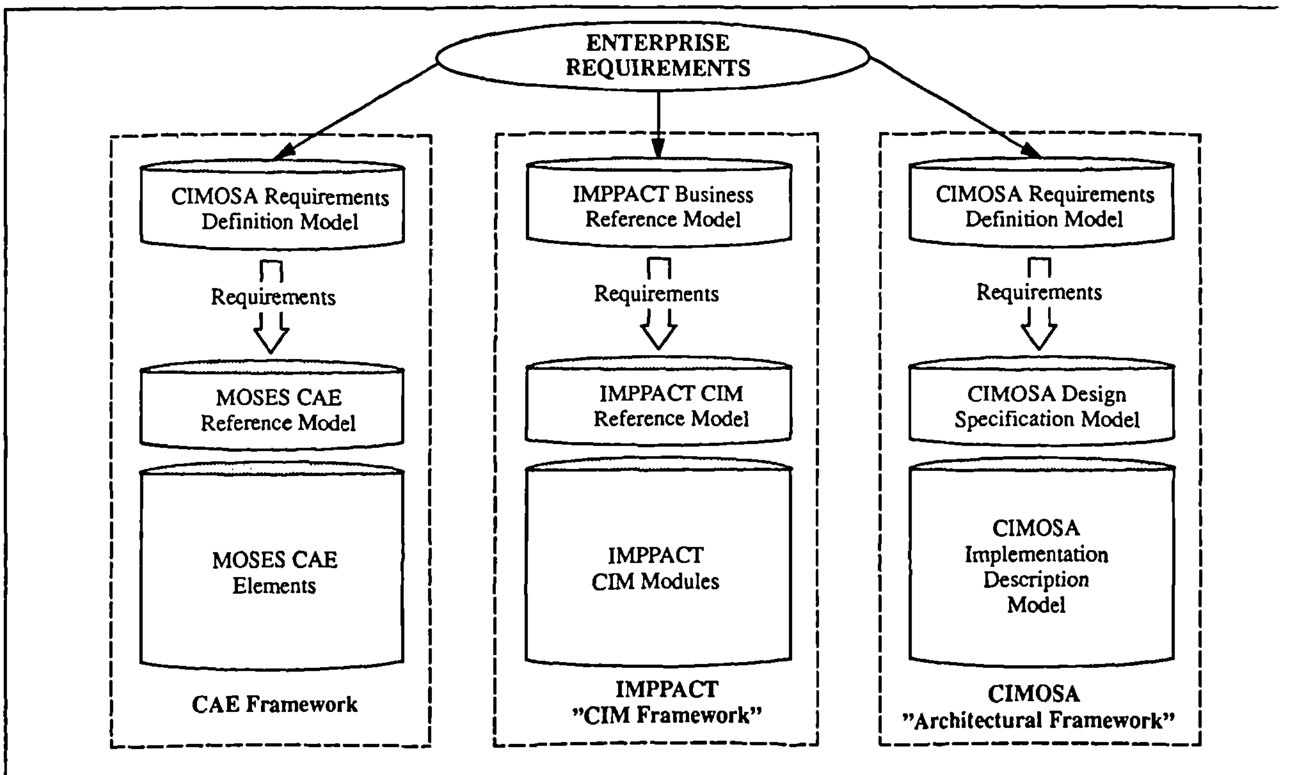
Once an agreement was reached, between the members of the MOSES research group, on the content of the first three levels of the IDEF0 model (A0, A1, A2 and sub–levels of A1 and A2, see appendix C for details). The task of developing IDEF0 models for each CAE system element (i.e. Manufacturing Model, Engineering Moderator and Design for Manufacture Environment) was assigned to the researcher responsible of its development. Therefore the author was responsible to develop the IDEF0 model for the Manufacturing Model. The complete IDEF0 model of the MOSES CAE–RM related to this PhD research is in appendix C, the IDEF0 model of the Manufacturing Model is in appendix D.

7.6. The relation between the CIMOSA Requirements Definition Model and the MOSES CAE Reference Model

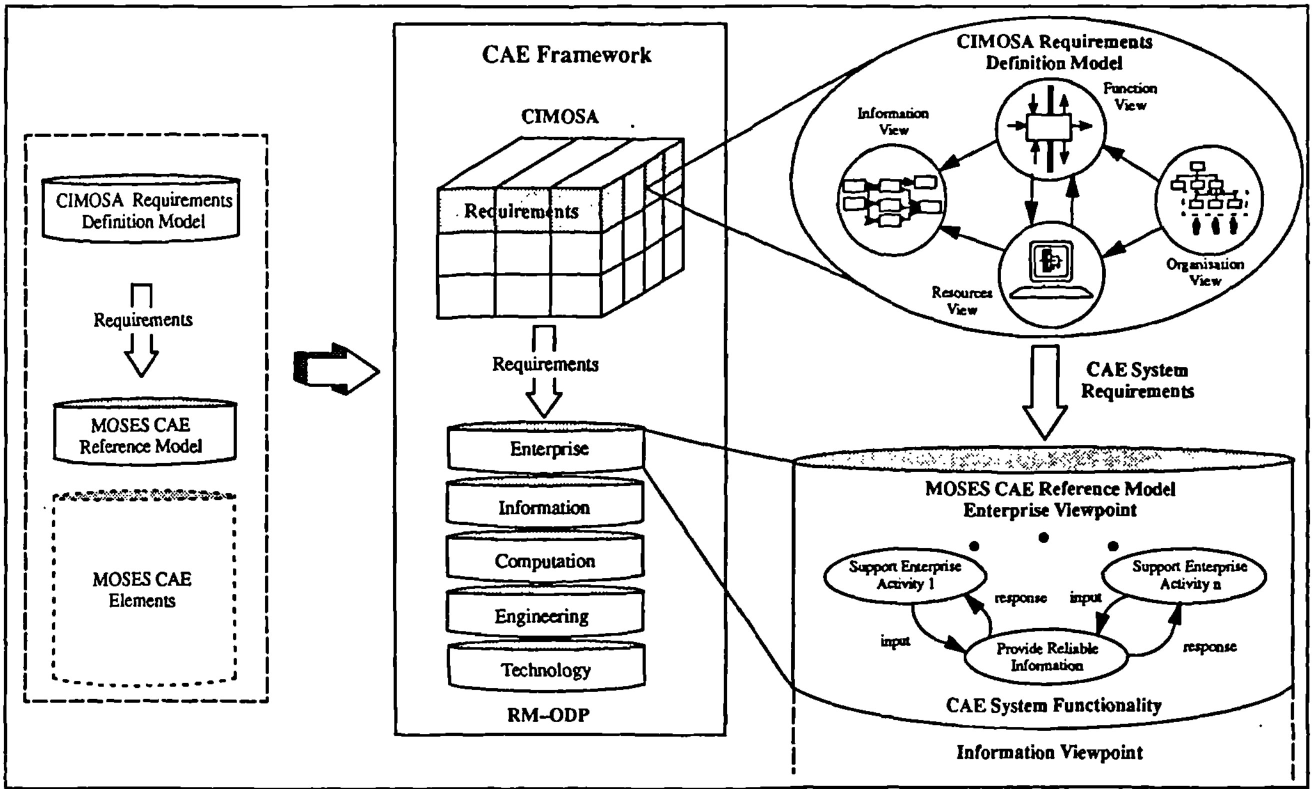
The Enterprise Viewpoint of the MOSES CAE Reference Model (CAE–RM) allows the establishment of the relation between the CIMOSA Requirements Definition Model and the the MOSES CAE–RM in the CAE Framework. This relation is created between an enterprise activity defined in the CIMOSA model and an activity defined in the IDEF0 model of the Enterprise Viewpoint of the MOSES CAE–RM. Figure 7.6 illustrates this point. Each enterprise activity is defined in terms of inputs and outputs. The CIMOSA resource inputs define the capabilities that are required from a resource in order to perform a certain activity. Thus the specific CAE system capabilities which are needed to realize certain enterprise activities are defined and from this definition the CAE system requirements are derived. The respective IDEF0 activity of the Enterprise Viewpoint which describes the function of the system that can satisfy that requirements is linked to the CIMOSA enterprise activity as depicted by the figure 7.6. The detailed exploration of this relation is presented in Chapter 8, Section 8.3.



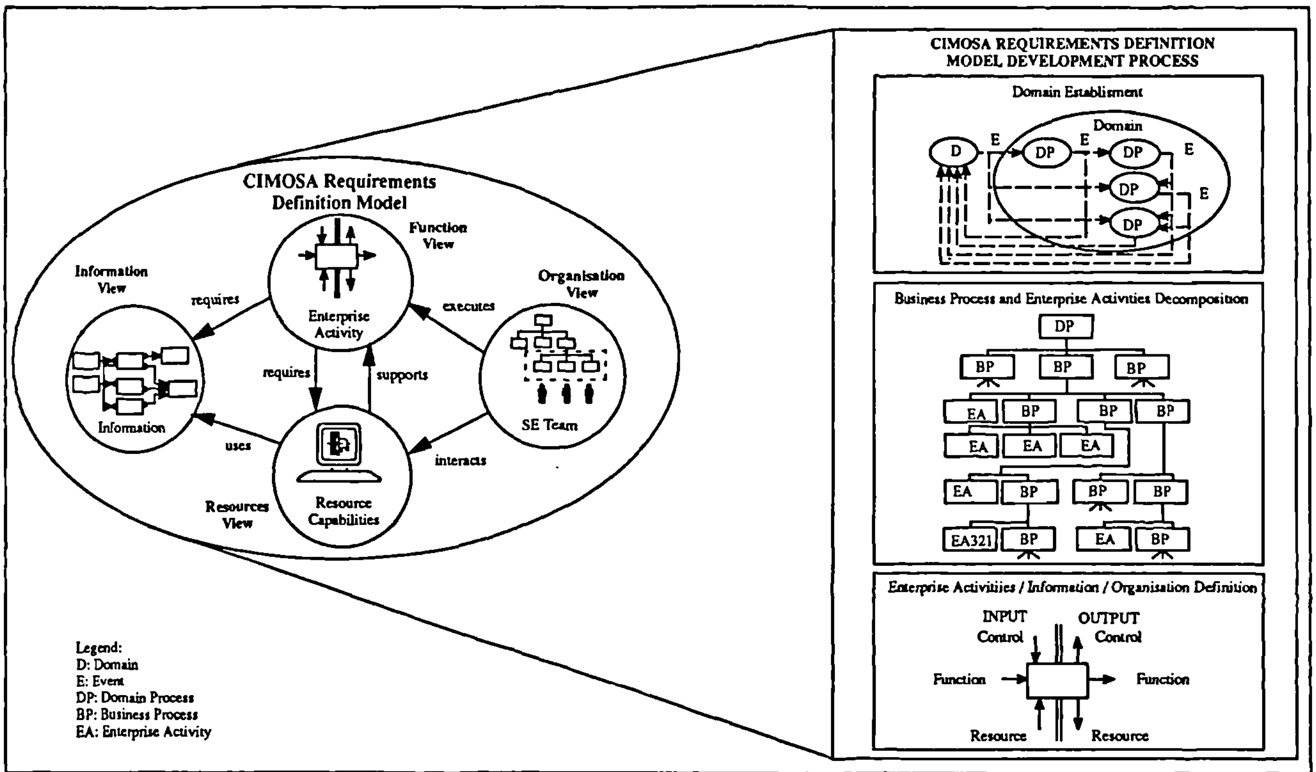
A. Molina	Figure 7.1 Enterprise Requirements as the Drivers for CAE System Selection	LUT–SE Research Group



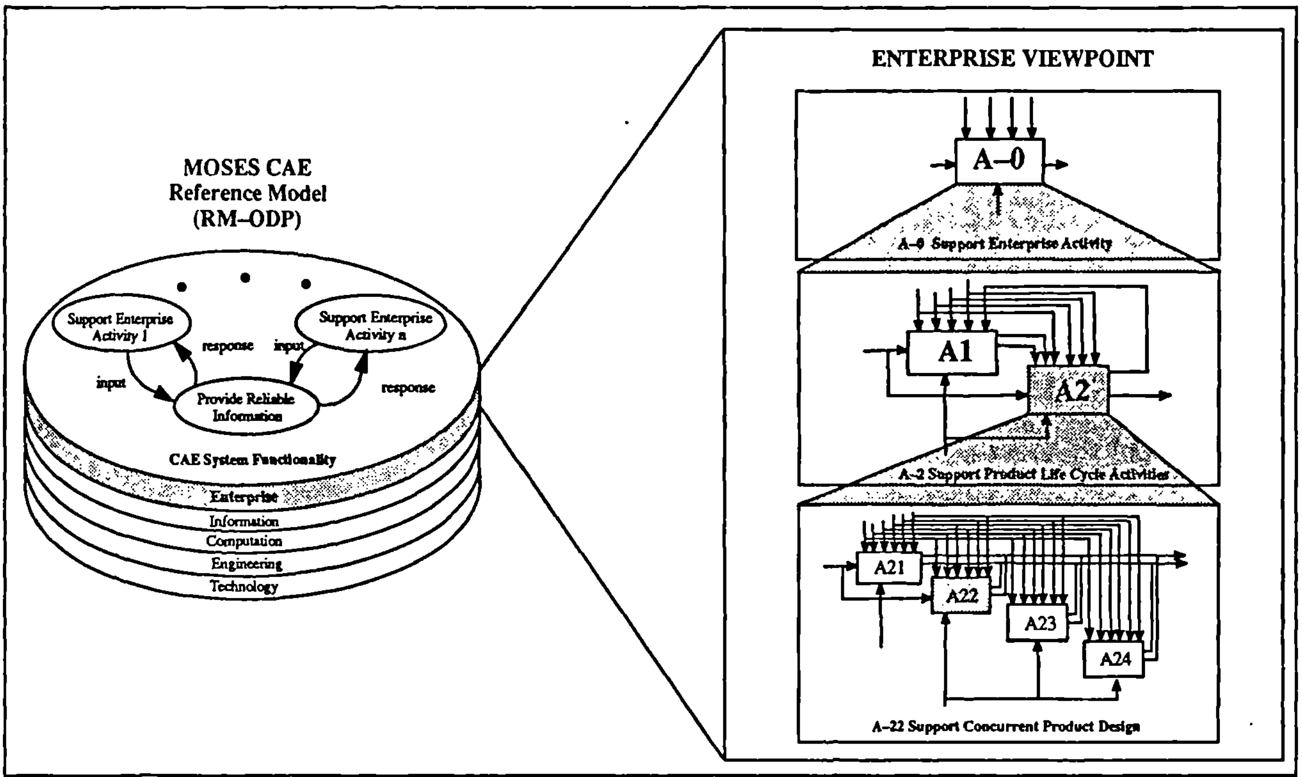
A. Molina	Figure 7.2 Competitive Frameworks for CAE System Development	LUT–SE Research Group



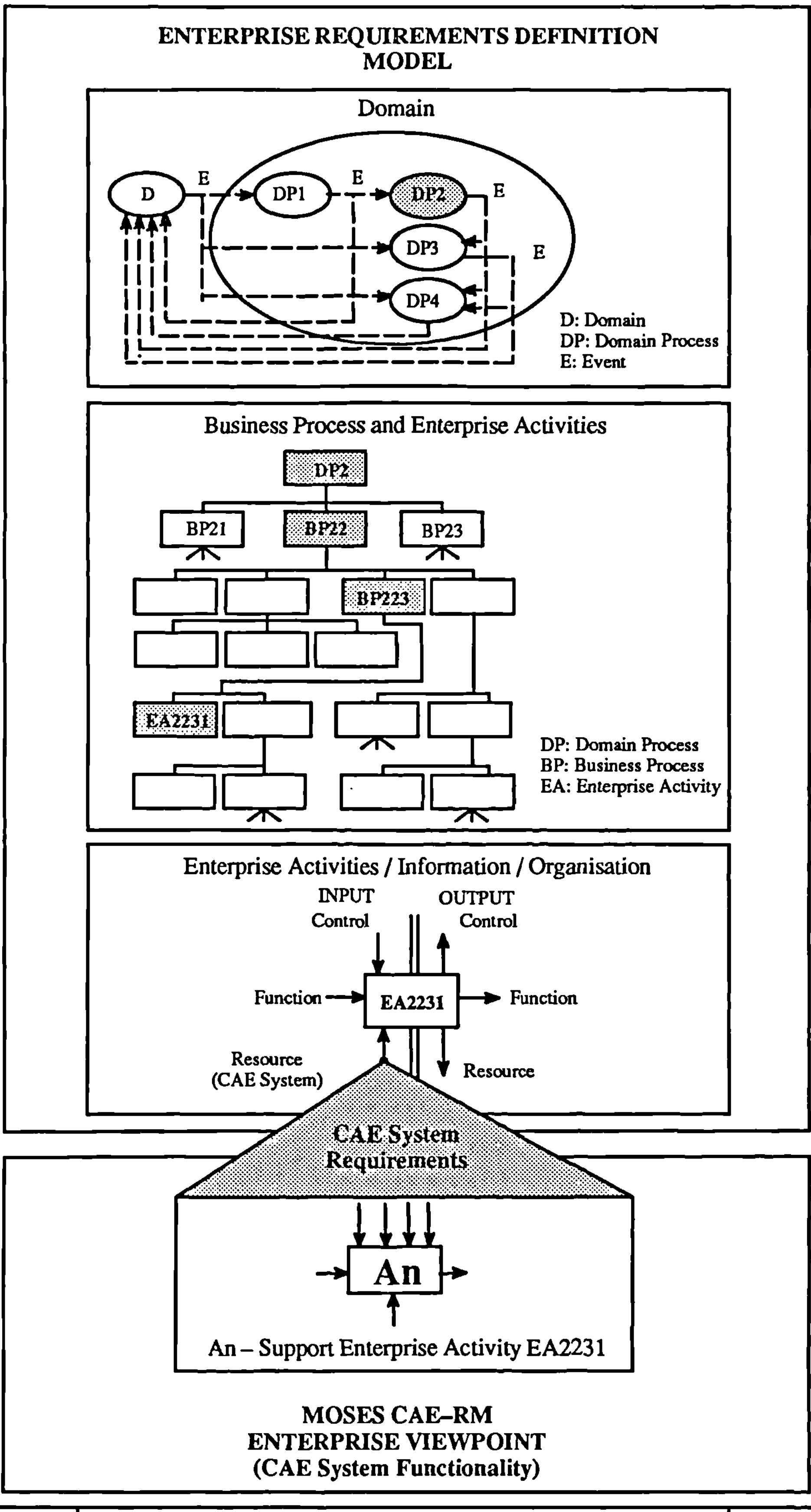
A. Molina	Figure 7.3 The Relation between the CIMOSA Requirements Definition Model and the Enterprise Viewpoint of the MOSES CAE Reference Model	LUT-SE Research Group



A. Molina	Figure 7.4 The development of the CIMOSA Requirements Definition Model in the CAE Framework	LUT-SE Research Group



A. Molina	Figure 7.5 The development of the Enterprise Viewpoint of the MOSES CAE Reference Model in the CAE Framework	LUT-SE Research Group



A. Molina	Figure 7.6 Establising the relation between the CIMOSA model and the MOSES CAE–RM	LUT–SE Research Group

Chapter 8

The use of the CIMOSA Requirements Definition Model in the CAE Framework

8.1. Introduction

This chapter describes the building process and decisions made by the author in order to create the CIMOSA Requirements Definition Model. This partial CIMOSA model describes the Concurrent Product Development Domain of an enterprise and enables the specification of the CAE System capabilities. These capabilities represent what is required from the CAE system in order to support the enterprise activities in the realization of concurrent product development. The definition of CAE system capabilities is used to derive the CAE system requirements which are the drivers to define the functions which are required from the CAE System in order to satisfy those capabilities. These functions are represented at the Enterprise Viewpoint of the MOSES CAE–RM. An example of how the requirements are satisfied by the functionality of the CAE system is presented in detail.

8.2. A CIMOSA Requirements Definition Model for Concurrent Product Development

The main objective in the development of the CIMOSA Requirements Definition Model in this research is to demonstrate how requirements for a CAE system to support simultaneous engineering can formally be described in the domain of Concurrent Product Development of an enterprise. The benefit of using this approach is that better system configurations can be created by matching real enterprise requirements against the functionality provided by a CAE system. The System Requirement Definition has the following phases (The author advises the reader to see Chapter 3 Subsection 3.3.8 where short definitions of the constructs used at each phase are presented):

1. Domain Establishment: definition of the Domain to be modelled and identification of the set of Domain Process.

2. Behaviour Analysis: functional decomposition of Domain Process(es) into Business Processes and Enterprise Activities which cooperate together to achieve the desired objectives.

3. Operational Analysis: definition of all Inputs and Outputs (Function, Control and Resource) of the Enterprise Activities as Object Views. This includes the required resource capabilities and the authorised responsibilities.

4. Information Analysis: capture the information required by an enterprise expressed in the operational analysis in an Enterprise Object structure and Information Elements.

5. Organisation Analysis: definition of Organisation Centers and Organisation Structure of the enterprise.

6. Consistency Checking: verification of a consistent definition of the different views (Function, Information, Resource and Organisation)

The Partial Requirements Definition Model developed in this thesis is the representation of the Domain "Concurrent Product Development" of a Manufacturing Enterprise. In general, CIMOSA Partial Requirements Definition Models are not complete, the level of detail defined in the model is left to the user. Thus, the author decided just to define a model which could describe the required basic functionalities and information to realize design of products using simultaneous engineering principles, and the capabilities of a CAE system to support this. Therefore, the model is not complete, and focuses only on the most important aspects of concurrent product development. The following sections discuss the author's experiences while developing the model. The complete, but partial, CIMOSA Requirements Definition Model is in appendix E.

8.2.1. Domain Establishment

In CIMOSA, Domains describe a part of an enterprise and its relationships with the outside environment from a high–level management perspective. A Domain is defined by identifying and establishing Domain Objectives/Constraints, Events, relations with other Domains, Domain Processes and, if any, Declarative Rules. Templates have been defined in CIMOSA to assist the developers in this task.

In order to establish the Domain of the CIMOSA Requirements Definition Model, from now on referred as CIMOSA model, the author created a schematic representation of the model (Figure 8.1). The author decided to focus only in a specific Domain which is "Concurrent Product Development", and particularly in the domain process "Concurrent Product Design", as depicted in figure 8.1. This Domain is composed by four Domain Processes : DP–01 "New Product Specification", DP–02 "Concurrent Product Design", DP–03 "Manufacture of Products", and DP–04 "Post–Manufacture Activities". Different Events can occur within the Domain: EV–01 "Customer Request", EV–02 "Specification Release", EV–03 "Product Design Release", EV–04 "Product Release" and EV–05 "Event Release", all of them allow the establishment of the relations between Domain Processes and show when a Domain Process can be trigger.

The Domain "Concurrent Product Realization" is triggered by the Event EV–01 "Customer Request" generated by an External Domain (e.g. Customer, Sales or Marketing) and by doing so the three Domain Process belonging to the "Concurrent Product Development" Domain can take place: DP–01 "New Product Specification", DP–03 "Manufacture of Products", or DP–04 "Post–Manufacture Activities". The reason for defining the Domain in this form that if the request from the customer is a new product, a specification for that product has to be made before the concurrent design of product take place. On the other hand, if the customer request is for an already produced product, the request is an order to manufacture that specific product. Finally, if the request is related to post–manufacture activities (e.g. maintenance or disposal) of the product, the relevant activity can be performed within the Domain. Other Events are generated at the end of the realization of a given Domain Process. For example, after the completion of DP–01 "New Product Specification", the Event EV–02 "Specification Release" is generated to enable the realization of the DP–02 "Concurrent Product Design". This Event can, as well, trigger the External Domain. If the External Domain was defined as Customer Domain, for example, EV–02 could represent the situation when the customer is asked to authorize a product specification.

Once the context of the Domain was understood, the author used the templates defined by CIMOSA to describe in detail the Domain, Objective/Constraints and Events. Figure 8.1 shows an example of the template for the Domain "Concurrent Product Development". The author found very useful the use of the templates for the description of the CIMOSA model because a systematic representation of the enterprise requirements can be made, and the templates ensure that a consistent model is built. At this stage not all the fields of a template can be filled, although the more information gathered the more meaningful the model becomes. The author has defined the objectives of the Domain "Concurrent Product Development" to be the ones commonly defined to be achieved when applying simultaneous engineering principles: reduce development time, improve quality and reduce cost, see figure 8.1. On the other hand, the constraints represented were related to the conformity with the business strategy and policies, the use of already available technology (e.g. Product Models), utilization of national and international standards (e.g. STEP/EXPRESS), and the employment of all experienced personnel. Finally, the most relevant Object Views were identified. The Object Views recognised by the author were: OV–01 "New Product order", OV–02 "Product order", OV–03 "Post–manufacture order", OV–04 "Product Specification", OV–05 "Product Design", OV–06 "Product". These Object Views are defined and associated to the Events in order that the Domain Process which is triggered can have access to this information. For example, when the EV–02 "Specification Release" is generated, the Object View OV–04 "Product Specification" is created and associated to the Event EV–02. This will enable that the Domain Process DP–02 "Concurrent Product Design" will have access to the "Product Specification".

The realization of this task was found to be very helpful to the author because the context where the modelling work has to be concentrated is clearly defined. The complete templates of the CIMOSA model can be found in appendix E.

8.2.2. Behaviour Analysis

The objective of this phase is to define the behaviour of Domain Processes. Nevertheless, the author considered that the complete description of the behaviour was not necessary for this research for the following reasons: the CIMOSA model does not need to be executed, the relevant aspects of the CIMOSA model, related to this research, are specified without the detailed description of behaviour, and the CIMOSA model fulfils the research objectives without the behaviour description. Therefore, the author in this phase concentrated his attention on the decomposition of the Domain Processes into Business Process and Enterprise Activities, defining Business Process Objectives/Constraints and Inputs/Outputs, describing Ending Status of Enterprise Activities, and creating the description template of a Domain Process. The activities related to the description of the behaviour of each Business Process and Domain Process were not carried out.

The decomposition of Domain Process into Business Processes and Enterprise Activities is the first step. The Domain Processs DP–02 "Concurrent Product Design" has been decomposed in three Business Processes: BP–2.1 "Management of Design", BP–2.2 "Design for Life Cycle", and BP–2.3 "Analysis and Test of Design", as depicted in Figure 8.2. For the purpose of this thesis, only the BP–2.2 "Design for Life Cycle" was decomposed further, and only the aspects related to design for function and design for manufacture were completed. Thus, the Business Process BP–2.2.3 "Create Embodiment Designs", together with its Enterprise Activities EA–2.2.3.1 "Do Design for Function" and EA–2.2.3.2.1 "Do Design for Manufacture" and Business Process BP–2.2.3.2 "Evaluate Embodiment Designs" were analysed (Figure 8.2).

Having finished the identification of Business Processes, one has to define the Objectives and Constraints for each Business Process, together with the Inputs and Outputs for

Function, Control and Resource. The definition of the Function Output allows the documentation, at this early stage, of the desired end result of a Business Process or Enterprise Activity. For example, the Function Output for the BP–2.2 "Design for Life Cycle" is a product design where the considerations for all its life cycle have been reflected in the design i.e. "Product_Design". On the other hand, the identification of what resources are required can be defined, such as the use of a "CAE System". In the same manner the information which acts as a control can be expressed as well, i.e. "Product_Specification". The complete set of templates for the Business Processes and Enterprise Activities are in appendix E.

8.2.3. Operational and Organisation Analysis

This third phase aims to define all Inputs and Outputs of the Enterprise Activities identified in a Domain Process. An Enterprise Activity is defined in terms of its inputs and outputs, see figure 8.3. The inputs and outputs can be of three types: function, control and resource. In this research this phase is the most important because the input/outputs definitions of the Enterprise Activities allows the derivation of the CAE system requirements,i.e. the information to be handled by the CAE system (function and control inputs) and CAE system capabilities (resource input).

The function input/output defines a set of Object Views which are processed by the functionality of that Enterprise Activity. For example in the Enterprise Activity EA–2.2.3.2.1 "Do Design for Manufacture" , illustrated in figure 8.3, the function input/output is the Object View OV–08 "Preliminary Product Design". This means that Object View OV–08 is the input and the output of EA–2.2.3.2.1 "Do Design for Manufacture". Even when the function input and output are defined by the same Object View (OV–08), the actual contents of the Object View (OV–08) may be different if modifications have been made to the product design after the activity "Do Design for Manufacture" has been performed. This is possible because there can be different instances of the same Object View. In the next subsection an extended explanation of Object Views is given. If a CAE system is going to support the Enterprise Activity

EA–2.2.3.2.1 "Do Design for Manufacture" it must be able to handle product information, such as the preliminary product design.

The control input contains information used to control or to constrain the execution of an Enterprise Activity, but not being processed by the Enterprise Activity. Following the same example of figure 8.3, the control inputs defined for the EA–2.2.3.2.1 "Do Design for Manufacture" are: OV–04 "Product Specification", OV–09 "Manufacturing Capabilities Information" and OV–10 "Design for Manufacture Rules (DFM)". Therefore in order to realize the activity "Do Design for Manufacture" it is necessary to apply DFM rules by considering the manufacturing capabilities and by complying at all times with the product specification. The control inputs represents other type of information which may be required to be manipulated by the CAE system, in particular information which constraints the Enterprise Activity. In this example the necessary information is related to the product (i.e. product specification), manufacturing facility (i.e. manufacturing capabilities information) and engineering knowledge (i.e. design for manufacture rules).

The control outputs provide a set of Events generated by the Enterprise Activity. These Events can trigger other Domain Process, for example the Event EV–06 "Design Change" could trigger a Domain Process in charge of monitoring product design changes. Even the description of events is not relevant for the author's research, it is important to consider them in order to build a more complete CIMOSA model.

The resource input describes the required resources (or set of resources) for the execution of an Enterprise Activity. The required capability of the resources needed to support the Enterprise Activity is documented using the template provided to define resource capability. The set of templates describing the capabilities of the resources constitutes the Resource View. In the example showed in figure 8.4 the required capability of a CAE system to support the activity "Do Design for Manufacture" has been specified at the Resource View. The description of resource capability is very important because it specifies what is needed of that resource, in this research this means, what is needed from the CAE system. The author has defined in the resource template (figure 8.4) the desirable characteristics of the CAE system. The CAE system should support the design for manufacture activity by giving design advice on process choice and manufacturability, and by providing a set of design for manufacturing rules. In addition, the CAE system must provide and handle information regarding the product being developed and the manufacturing facility capability where the product will be produced. No resource output is defined as there is no information to be recorded on the usage of the CAE system after the execution of this Enterprise Activity.

With the definition of the resource capabilities the operational analysis is concluded. The next stage is to define the organisational aspects of the model. These aspects are related to the definition of the responsibilities (person or organisational unit) for carrying out the Enterprise Activities and to create and maintain the CIMOSA model. This is represented in the Organisation View, and is described in the templates by defining the "Design Authority" item. In the CIMOSA model of this research the Simultaneous Engineering Team is the design authority. The author would like to point out that this view is not well defined in the CIMOSA documentation and therefore the Organisation View was not developed further.

At this stage all the Enterprise Activities of the CIMOSA model have been fully described and documented using the CIMOSA templates. The next phase is very important for this research because the definition of the information requirements to realize the Enterprise Activities has a major influence on determining the necessary information to be modelled in the Information View. Thus the information which will be required to be captured in the Manufacturing Model.

8.2.4. Information Analysis

The purpose of this phase is to capture the essential information required to support the execution of the Enterprise Activities defined in the Function View. The information has been described in the Enterprise Activities as Objects Views. In the example illustrated figure 8.5 the Object Views are : OV–04 "Product Specification", OV–08 "Preliminary Product Design", OV–09 "Manufacturing Capabilities Information" and OV–10 "Design for Manufacture Rules". The results of the information analysis should be a description of Objects Views, Information Entities and Enterprise Objects (The author would like to remind again the reader to see Chapter 3, Subsection 3.2.8 for the definition of these concepts). Nevertheless, the information analysis carried out by the author focuses only on defining the Object Views because these object descriptions represent the type of information required to support an Enterprise Activity. In figure 8.5, the information required to support the realization of the EA–2.2.3.2.1 "Do Design for Manufacture" related to the OV–09 "Manufacturing Capability Information" is, among other: type and configuration of resources and facilities, type of process, physical capabilities of resources and processes, and attributes related to cost, times and availability. It is important to mention that the definition of manufacturing capabilities is constrained by the companies' strategic decision and operational rules. The author decided that no further decomposition was necessary, therefore the CIMOSA model does not include the definitions about Information Elements or Enterprise Objects. This decision was made based on the fact that by defining only Object Views no restrictions are imposed on the structure and content of the Manufacturing Model developed in this research. Although the type of information required to be captured by the Manufacturing Model has been defined by the properties of the Object View OV–09 "Manufacturing Capability Information".

This analysis concludes the development of the CIMOSA Requirements Definition Model, although the model is not complete, the model captures, in the author's opinion, the relevant elements needed in order to derive the CAE system requirements, which are: activities to support and information to be provided. The next section explains how the relation between the CIMOSA model and the Enterprise Viewpoint of the MOSES CAE Reference Model was made for the Enterprise Activity "Do Design for Manufacture".

8.3. The exploration of the relation between the CIMOSA Requirements Definition Model and the Enterprise Viewpoint of the MOSES CAE Reference Model

The CAE system requirements derived from the CIMOSA Requirements Definition Model specify the following:

1. What kind of support is expected from the CAE system, described by the resource capability

2. What type of information is required to be handled by the CAE System, described by the Object Views.

In the Enterprise Activity "Do Design for Manufacture" the following CAE system capabilities has been defined to be required from a CAE System (figure 8.4):

- Give advice on process choice
- Give advice on manufacturability
- Provide a set of design for manufacturing rules
- Provide product information (e.g. specification, geometry, manufacturing data) and manufacturing infoımation (e.g. manufacturing capability of a process, resource or facility).

and the following information is required to be handled (figure 8.5):

- Product Information: product design and product specification
- Manufacturing Capability Information

Therefore the CAE system should provide the necessary functions to be able to satisfy these requirements. The Enterprise Viewpoint of the MOSES CAE Reference Model represents the functions provided by the CAE system and its elements (e.g. Product Model, Manufacturing Model, Design for Manufacture Environment, Engineering Moderator) in order to satisfy the requirements of CIMOSA Requirements Definition Models (This point was discussed in Chapter 7 Section 7.5). By doing so, all the functions provided by the CAE system elements support enterprise activities, and therefore, the IDEF0 activities representing functions of the CAE system could be linked directly to an Enterprise Activity. In other words, for each Enterprise Activity of the CIMOSA model, where the CAE system is required, a function or functions to provide such support are defined in the Enterprise Viewpoint (Figure 8.6).

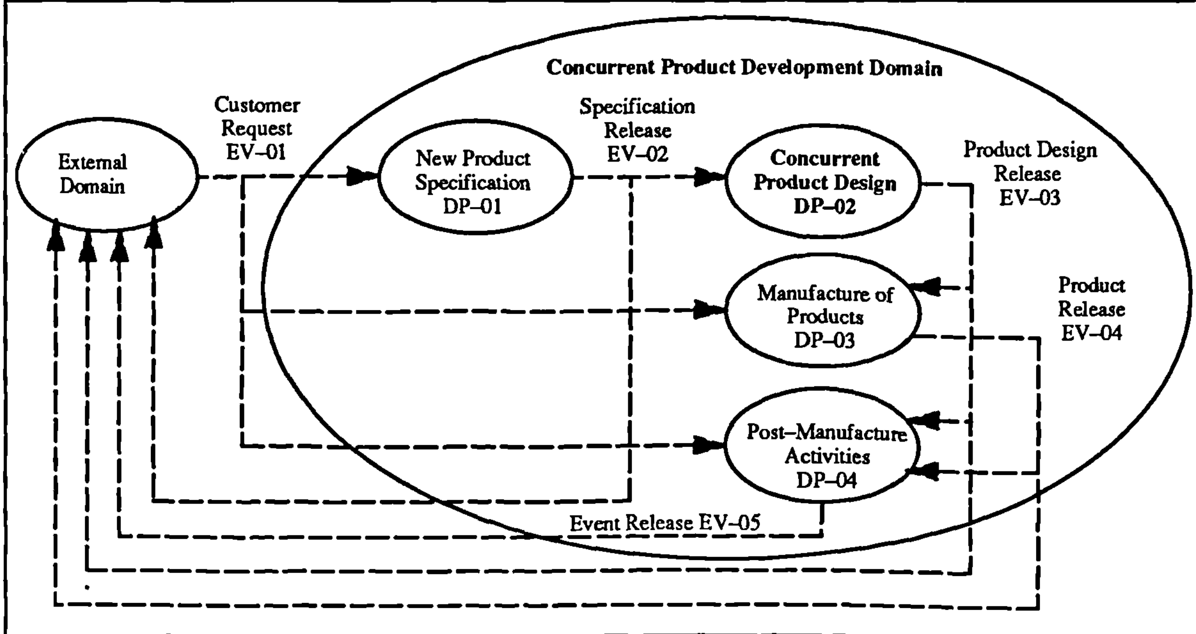
The figure 8.6 illustrates this argument. The following two CAE system functions support the Enterprise Activity "Do Design for Manufacture":

- The function "Support Design for Manufacture" is responsible to give advise on process choice and manufacturability based on design for manufacturing rules. In order to perform such function, the element required from the CAE system is the Design for Manufacture Environment, specified in the IDEF0 definition as a resource (mechanism). The control inputs applied to the Enterprise Activity "Do Design for Manufacture" are the same for the function "Support Design for Manufacture", i.e. Product Specification (Product Information), Manufacturing Capability Information (Manufacturing Information), DFM Rules (DFM Rules) which are controls of the IDEF0 representation. As the IDEF0 diagram represents the functionality of the CAE System, the input not necessarily is the function input "Preliminary Product Design", but at certain stages of the usage of the CAE system it could represent modifications made to the product design. On the other hand, the function output "Preliminary Product Design" is represented by the Product Information Updated if any changes were made to the product design.

- The function "Provide Reliable Information Throughout the Product Life Cycle" is responsible for providing the product and manufacturing information required by the function "Support Design for Manufacture". The elements of the CAE system which are used as the mechanisms to procure this set of information are the Product and Manufacturing Model.

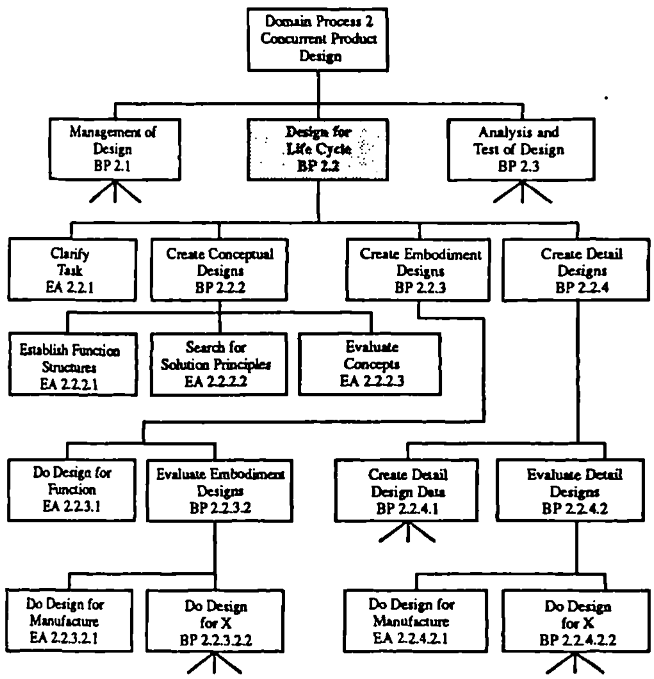
All the constraints applied to the CIMOSA model are reflected in the Enterprise Viewpoint, i.e. Available Technology, International and National Standards, Engineering Knowledge, etc.

The concept presented here is very important because different enterprises can developed their own CIMOSA Requirements Definition Models by specifying the Enterprise Activities which requires CAE system support. By defining the requirements for those Enterprise Activities different CAE system configurations can be built using the MOSES CAE–RM. These configurations will represent various CAE system functions and related CAE system elements which satisfy the specific needs of those Enterprise Activities.



DOMAIN	
Part 1:	DOMAIN DESCRIPTION
TYPE:	tbd
IDENTIFIER:	DM–01
NAME:	Concurrent Product Development
DESIGN AUTHORITY:	Simultaneous Engineering Team
DOMAIN DESCRIPTION:	Concurrent Product Development is a Domain assigned to design, manufacture and support a product throughout its life–cycle using simultaneous engineering principles and supported by a Computer Aided Simultaneous Engineering System (CAE System).
CIMOSA COMPLIANT:	yes
Part 2:	DOMAIN COMPONENTS
DOMAIN OBJECTIVES:	DO–01/reduce product development time DO–02/improve quality of product DO–03/reduce cost of product
DOMAIN CONSTRAINTS:	DC–01/Business Strategy and Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DOMAIN PROCESSES:	DP–01/NEW PRODUCT SPECIFICATION DP–02/CONCURRENT PRODUCT DESIGN DP–03/MANUFACTURE OF PRODUCTS DP–04/POST–MANUFACTURE ACTIVITIES
DOMAIN BOUNDARY:	RL–01/External Communication
OBJECT VIEWS:	OV–01: New Product order OV–02: Product order OV–03: Post–manufacture order OV–04: Product Specification OV–05: Product Design OV–06: Product
EVENTS:	EV–01: Customer request EV–02: Specification Release EV–03: Product Design Release EV–04: Product Release EV–05: Event Release

A. Molina	Figure 8.1 Establishing the Domain "Concurrent Product Development"	LUT–SE Research Group

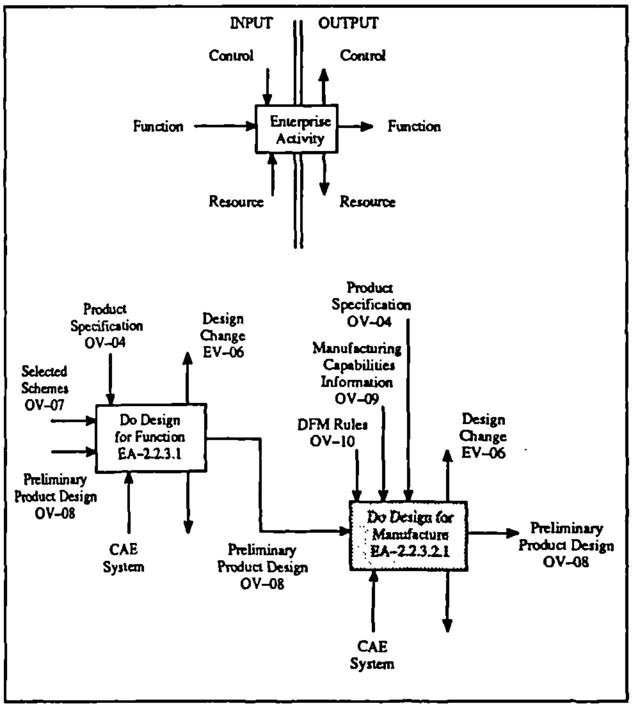


BUSINESS PROCESS: BP-2.2, DESIGN FOR LIFE CYCLE	
TYPE:	tbd
IDENTIFIER:	BP-2.2
NAME:	DESIGN FOR LIFE CYCLE
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO-01/reduce product design time DPO-02/improve quality of product DPO-03/reduce cost of product design
CONSTRAINTS:	DC-01/Design Policies DC-02/Available Technology DC-03/National and International Standards DC-04/Engineering Knowledge
DECLARATIVE RULE:	tbd
DESCRIPTION:	Business Process designs products while considering all the aspects of their life-cycle using simultaneous engineering principles. The Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	tbd
FUNCTION OUTPUTS:	Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	tbd
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	EA-2.2.1: Clarify Task BP-2.2.2:Create Conceptual Designs BP-2.2.3:Create Embodiment Designs BP-2.2.4:Create Detail Designs
PROCEDURAL RULES:	tbd

A. Molina

Figure 8.2 Decomposition of the Domain Process "Concurrent Product Design" in Business Processes and Enteprise Activities

LUT-SE Research Group

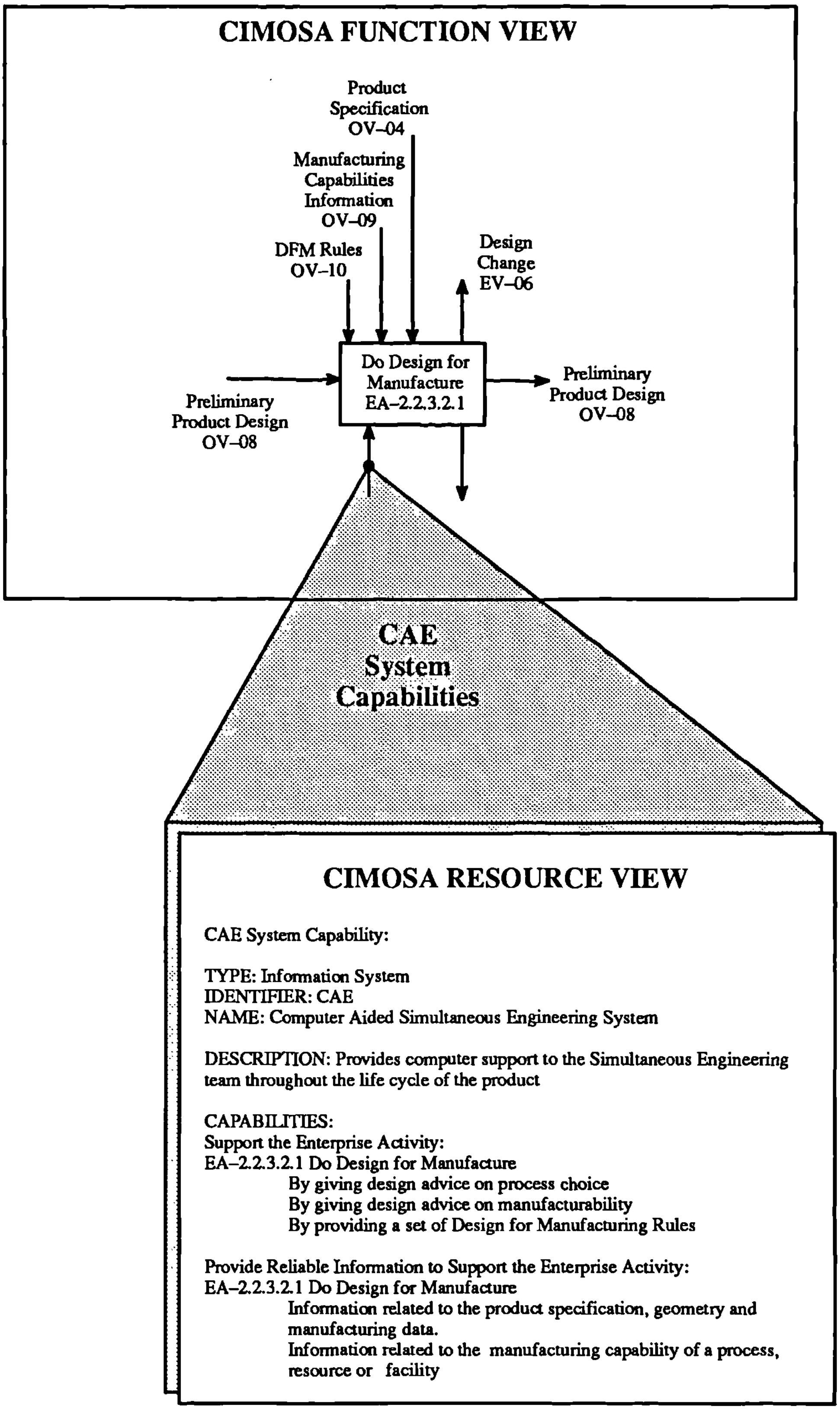


ENTERPRISE ACTIVITY: EA-2.2.3.2.1, DO DESIGN FOR MANUFACTURE	
TYPE:	tbd
IDENTIFIER:	BP-2.2.3.2.1
NAME:	DO DESIGN FOR MANUFACTURE
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO-01/reduce product design time DPO-02/improve quality of product DPO-03/reduce cost of product design
CONSTRAINTS:	DC-01/Design Policies DC-02/Available Technology DC-03/National and International Standards DC-04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Apply Design for Manufacture Rules to the Preliminary Product Design to evaluate how the way in which a product will be manufactured influences the product design. Design changes might be suggested in order to improve the product design. This Enterprise Activity is supported by a Computer Aided Simultaneous Engineering System (CAE System).
INPUTS	
FUNCTION INPUT:	OV-08/Preliminary_Product_Design
CONTROL INPUTS:	OV-04/Product_Specification OV-09/Manufacturing Capabilities Information OV-10/Design for Manufacture (DFM) Rules
RESOURCE INPUTS:	CAE System
OUTPUTS:	
FUNCTION OUTPUTS:	OV-08/Preliminary_Product_Design
CONTROL OUTPUTS:	EV-06/Design_Change
RESOURCE OUTPUTS:	none
ENDING STATUS:	completition status: done (preliminary design)
REQUIRED CAPABILITIES:	CAE System

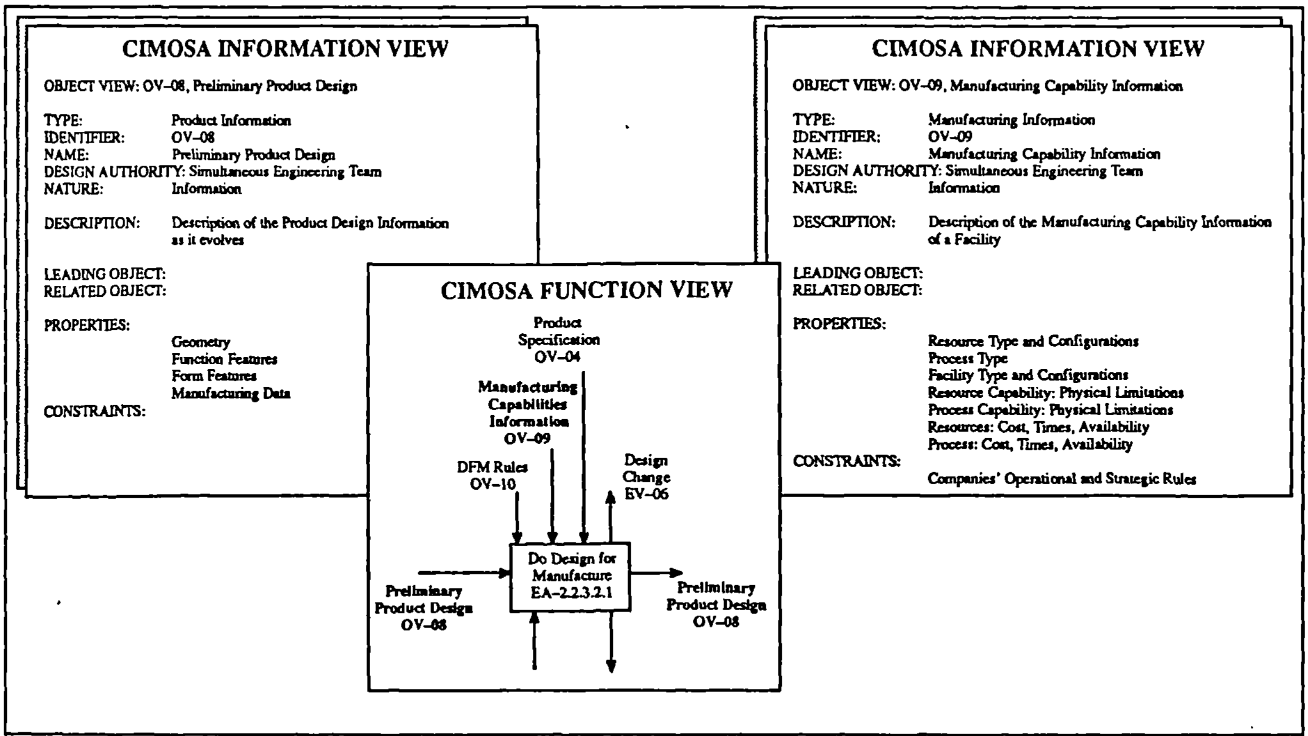
A. Molina

Figure 8.3 Definition of the Enterprise Activity "Do Design for Manufacture"

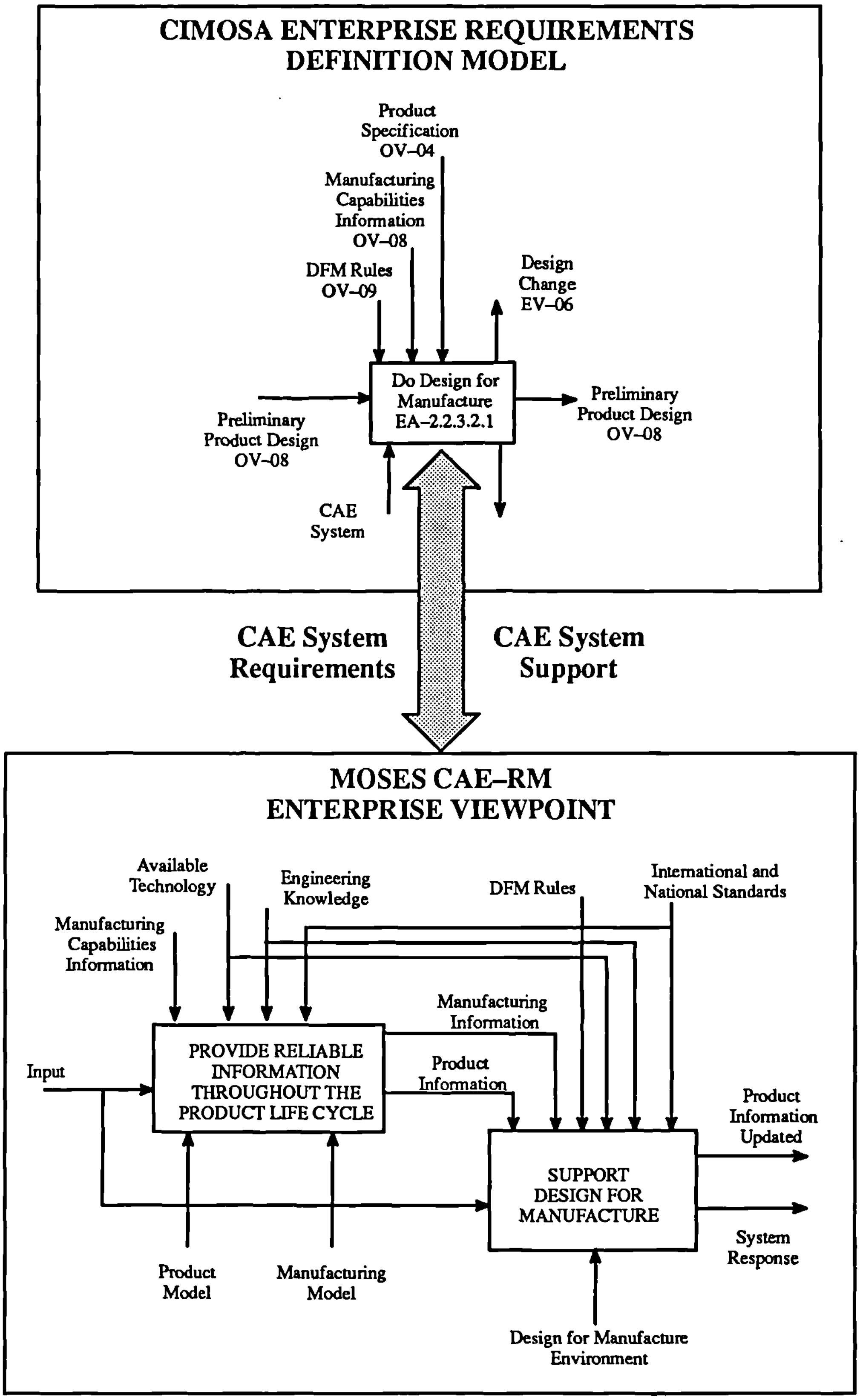
LUT-SE Research Group



A. Molina	Figure 8.4 CAE System Capabilities Described in the Resource View	LUT–SE Research Group



A. Molina	Figure 8.5 Description of the Object Views "Preliminary Product Design" and "Manufacturing Capability Information" at the Information View	LUT-SE Research Group



A. Molina	Figure 8.6 Linking Enterprise Activity "Do Design for Manufacture" to functions "Provide Reliable Manufacturing Information" and "Support Design for Manufacture"	LUT–SE Research Group

Chapter 9

The use of RM–ODP in the CAE Framework

9.1. Introduction

This chapter introduces key decisions made by the author in the implementation of the Reference Model for Open Distributed Processing (RM–ODP) as the basis for the MOSES CAE–RM. The description of the different viewpoints is presented in order to define the frame of reference for the Manufacturing Model development. The issues which have to be considered in the development of the Manufacturing Model to be consistent with the MOSES CAE–RM are presented.

9.2. The Specification of the MOSES CAE Reference Model

The MOSES CAE–RM is based on the RM–ODP. The use of the RM–ODP, in the author's opinion enforces the generic and modular characteristic of the MOSES CAE–RM. These characteristics are important in order to define a model independent of any existing applications and open–ended in its ability to be extended in order to incorporate new concepts and technologies. The RM–ODP is a five level model which is intended to represent open distributed systems. To achieve this, the following five levels have been defined: Enterprise, Information, Computation, Engineering, and Technology (Figure 9.1).

The Enterprise Viewpoint is associated with the specification of the CAE system functions. Nevertheless, the scope of the information described, and the level of detail contained in the Enterprise View remains a point of debate within the MOSES research group. Some researchers believe that the Enterprise Viewpoint should represent the complete enterprise, the author strongly disagrees with this argument. The author has agreed, with other members of the MOSES LUT research project, to represent at the Enterprise Viewpoint the functionality that the CAE system is intended to achieve in order to support the different enterprise activities described in the requirements definition model defined by the CIMOSA model. This argument enables the Enterprise View to be the link between the two reference models i.e. CIMOSA and RM–ODP. The concept behind this is to design CAE Systems based on the MOSES CAE–RM which could match the requirements of CIMOSA Requirements Definition Models. This concept has been presented in detail in Chapter 7 (Section 7.5 and Section 7.6), and has been explored in Chapter 8 (Section 8.3).

This Enterprise Viewpoint has been documented in IDEF0. The detailed explanation of why IDEF0 has been chosen to represent this viewpoint is in Chapter 10, Subsection 10.3.1. In this research, a team based project, this viewpoint established a common understanding of concepts and terminology. For the development of this viewpoint several meetings were held by the MOSES LUT Research group in order to understand better and define its content. The complete Enterprise Viewpoint of the MOSES CAE–RM related to this PhD research is in appendix C.

The Information Viewpoint focuses on describing the semantics of information and information processing functions in the CAE system. In the MOSES CAE–RM this viewpoint is defined via a combination of IDEF0 and EXPRESS models. This combination of models allows the description of the information flows together with the structure of the information elements, their relationships and quality attributes. The information viewpoint sets the context for the development of the information models required in the CAE system in an implementation independent form. Therefore this viewpoint is particularly important to the development of the Manufacturing Model. In Chapter 10, Subsection 10.3.2 the arguments on the selection of IDEF0 and EXPRESS to represent this viewpoint are presented.

The Computational Viewpoint focuses on the representation of the functional decomposition of the system into objects, the activities that occur within those objects and the interactions between the objects. The Booch method satisfies the requirements needed for the description of this view. The reasons for the use of Booch to describe this viewpoint are discussed in Chapter 10, Subsection 10.3.3 The MOSES CAE system comprises the following objects: Information Models (i.e. Product and Manufacturing Models), Design for X Application Environments (i.e. Design for Function, Design for Manufacture), Engineering Moderator, and Manufacturing Information Generation Environment.

The last two viewpoints (Engineering and Technology) are out of the scope of this PhD research, nevertheless a short explanation is offered. The Engineering Viewpoint focuses on the infrastructure required to support distribution. This view enables the specification of the processing, storage and communication functions required to implement the system. This viewpoint is supported by defining an Integration Environment which supports remote procedure call functions. In the author's opinion the use of CIMOSA integrated infrastructure can be explored as the basis for the definition of this viewpoint in order to enable the integration of the MOSES CAE System with other CIMOSA based systems. Finally, the Technological Viewpoint focuses on the selection of the necessary technology to support the system. In the MOSES research project, the object oriented database DEC Object/DB (Objectivity/DB 1991) and object oriented programming language C++ (Stroustrup 1986) were selected. A variety of hardware platforms including Sun Sparcstations and Dec Alphas have been employed.

The top levels of the reference model (enterprise, information and computation) are non–software specific and so they provide a base level description for system development. Although the RM–ODP allows the thorough description of a CAE system from different views, the author's research was primarily involved with the definition of the first three viewpoints i.e. Enterprise, Information and Computation. The reasons to focus on these three viewpoints are stated in the following section.

9.3. The Manufacturing Model within the MOSES CAE Reference Model

The MOSES CAE–RM is the frame which provides the means to support the development of the CAE system and its elements based on the generic set of viewpoints of the RM–ODP. The author has contributed to the definition of what methodologies and tools are being used to assist in the specification, development and analysis of each view (e.g. IDEF0, EXPRESS, Booch). This will ensure that certain key issues are considered during the design of the CAE system elements, and that standardised methods are used for the design and documentation of the CAE system.

Important element of the CAE–RM is the use of information models as sources of data to support the information needs for life cycle engineering activities. The Manufacturing Model is an information model which has to be defined independently from any application, designed according to the guide-lines provided by the CAE–RM, and implemented as an information system that could be integrated into any major open distributed system. The use of the CAE–RM allows the achievement of these goals. From the five viewpoints defined in the RM–ODP, the author is mainly concerned with the first three viewpoints: Enterprise, Information and Computation (Figure 9.2). The thorough definition of these viewpoints for the Manufacturing Model research, in the author's opinion, satisfies the requirements imposed in the Manufacturing Model in order to be generic, structured, modular and software independent. These characteristics allow the Manufacturing Model to be integrated easily into the complete MOSES CAE system.

One important characteristic of the RM–ODP, exploited by the author, was the possibility to define each element of information systems from the five ODP viewpoints. This characteristic gives a great modularity to the MOSES CAE–RM, as elements can easily be added, modified or deleted. The integrity and consistency between the CAE elements can be ensured by using the MOSES CAE–RM as the frame of reference. Integrity and consistency can be checked, at the Enterprise Viewpoint, by complying with the controls and inputs/outputs of the IDEF0 model. It is important, however, to conform with the definition given in the MOSES CAE–RM at each viewpoint, for every element defined and integrated into the CAE system.

The author has exploited this characteristic to build and define the Manufacturing Model using the five ODP viewpoints, and the methodologies defined in the MOSES CAE–RM.

Therefore, in the Manufacturing Model the Enterprise Viewpoint has to be documented in IDEF0, the Information Viewpoint in IDEF0 and EXPRESS, and the Computation Viewpoint in Booch notation.

The Enterprise Viewpoint is associated with the functionality of the Manufacturing Model which satisfies the requirements for the enterprise manufacturing information i.e. the Manufacturing Model should provide reliable manufacturing information for the realization of enterprise activities. According to the MOSES CAE–RM, IDEF0 activity diagrams are employed to describe this view.

The description of the Information Viewpoint of the Manufacturing Model is the major emphasis of this research work. A combination of IDEF0 diagrams and EXPRESS definitions are used for this purpose (Figure 9.2). The IDEF0 model of this Information Viewpoint is presented in detail in appendix F, and the EXPRESS model is described in some extend in Chapter 12.

The Computational Viewpoint is represented using Booch notation. The Manufacturing Model comprises the following objects: Information Model, Information Model Manager and Information Model Interface (Figure 9.2). These objects are required to design and implement the Manufacturing Model as an object oriented information system with the characteristics of being modular, flexible and open. The Computational Viewpoint of the Manufacturing Model, i.e. its design and implementation is explained in appendix J.

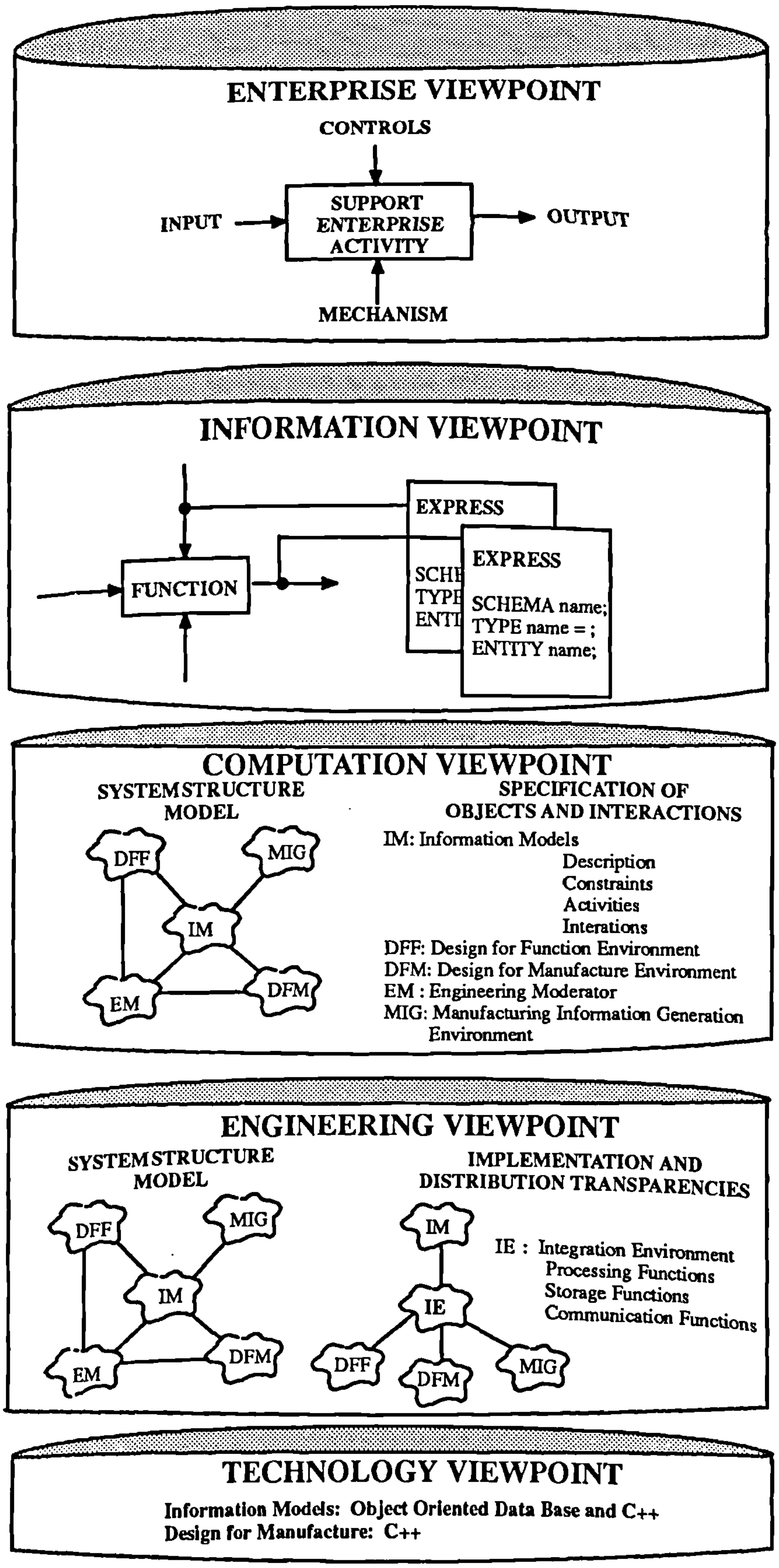
In order to remain consistent with these definitions, the A12 diagram of the IDEF0 model of the CAE–RM Enterprise Viewpoint becomes the top IDEF0 diagram A–0 of the Enterprise View for the Manufacturing Model (Figure 9.3). Then, all the necessary sub–levels can be developed to describe in a structured manner the different supporting functions of the Manufacturing Model. The main objective of this modelling exercise was to identify the range of information which was required to be represented in the Manufacturing Model in order to support the realization of certain activities. For example, the information needed to support Design for Manufacture activities includes process capabilities, machine capabilities, tools, etc. Some of this information was inferred from the CIMOSA Requirements Definition Model, as the requirements for information are defined in the input/output functions and input/output controls. The complete Enterprise Viewpoint of the Manufacturing Model is in appendix D.

9.4. The relation between the Enterprise Viewpoint and Information Viewpoint in the MOSES CAE Reference Model

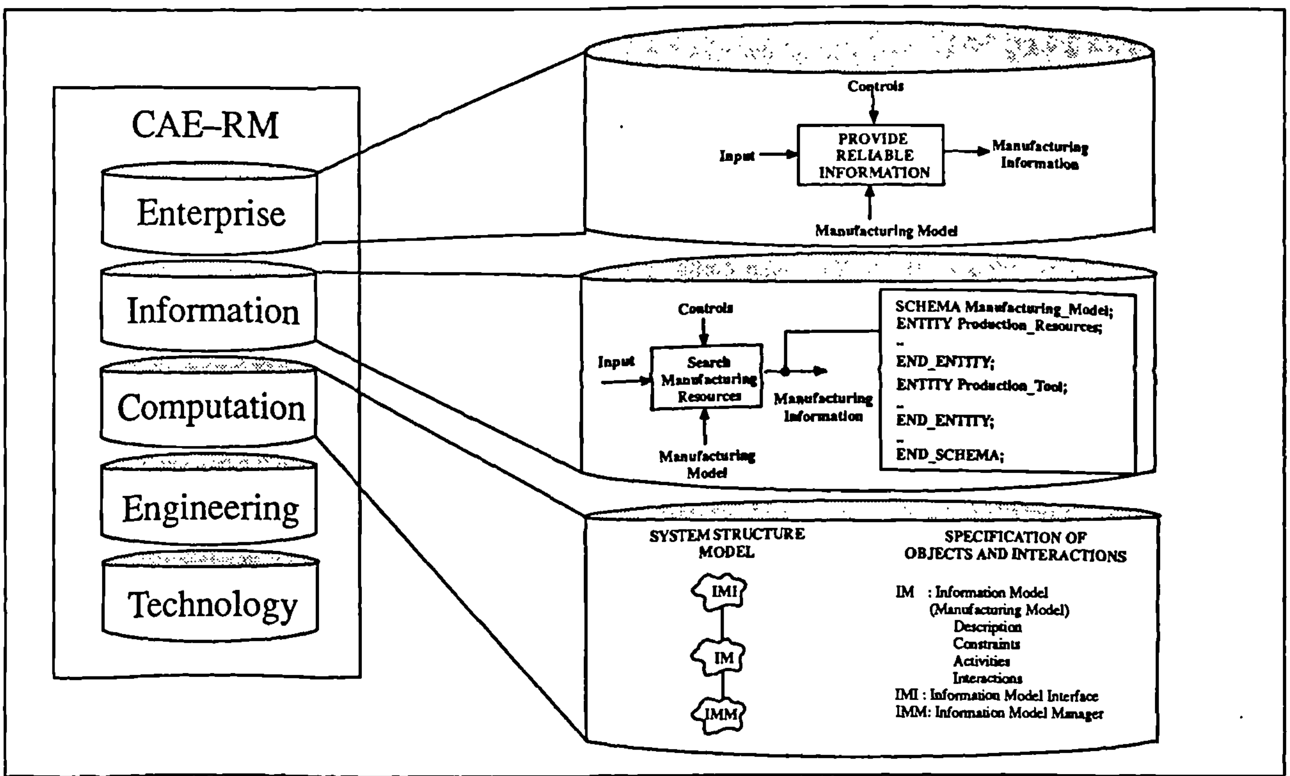
An important issue, discussed in the MOSES research group, was to make a clear distinction between what the IDEF0 models represent at the Enterprise Viewpoint and what the IDEF0 models represent at the Information Viewpoint. This is important because each researcher should know and decide how detailed the IDEF0 model at the Enterprise Viewpoint should be and how this model can be mapped to the Information Viewpoint. The decision made was that: at the Enterprise Viewpoint the IDEF0 models should represent what functions the CAE system provides, at any level of detail, and at the Information Viewpoint how these functions are implemented. The mapping between the Enterprise Viewpoint and Information Viewpoint is explained in the next section.

The mapping between the Enterprise Viewpoint and the Information Viewpoint is very important to ensure that the functionality can be achieved. This is done in the Manufacturing Model by associating the activities defined at the lowest sub–levels of the IDEF0 at the Enterprise Viewpoint with the IDEF0 model of the Information Viewpoint, as depicted in Figure 9.4. The IDEF0 model of the Information Viewpoint is a derivation of the IDEF0 diagrams of the Enterprise Viewpoint, describing how a specific function is implemented. In figure 9.4, for example, the function "Provide Reliable Manufacturing Information to support Design for Machining" (A211) can be implemented by performing the following activities "Search for Manufacturing Information" (A211–2), and then "Search for Manufacturing Process Information" (A211–22). This is true if the information required is related to the manufacturing process, but if the information required is about manufacturing resources, then the sequence stated above is not adequate, instead the following sequence should be applied

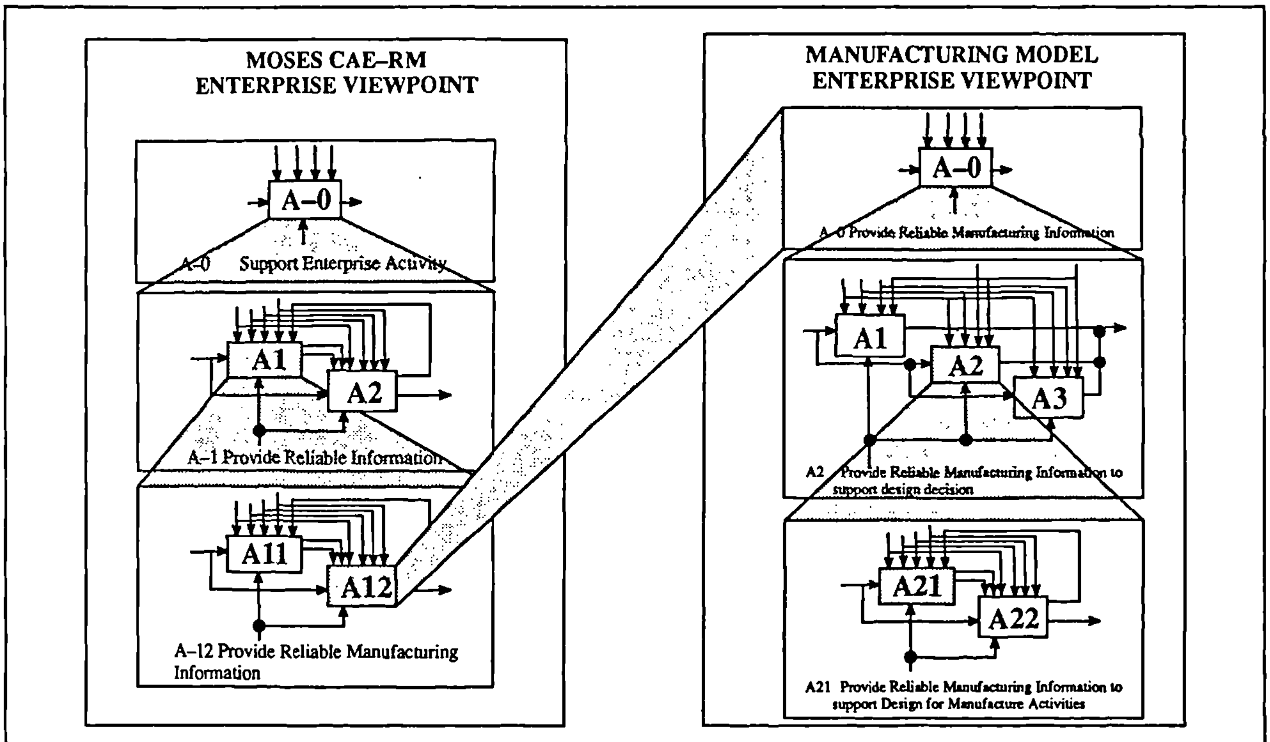
"Search for Manufacturing Information" (A211–2), and then "Search for Manufacturing Resource Information" (A211–21). To solve this problem the author found that there are a set of generic activities which are always carried out to implement almost all the Manufacturing Model functions. The IDEF0 model of the Information Viewpoint represents this set of activities which are common to implement all the *functions* provided by the Manufacturing Model. These activities are described in appendix F.



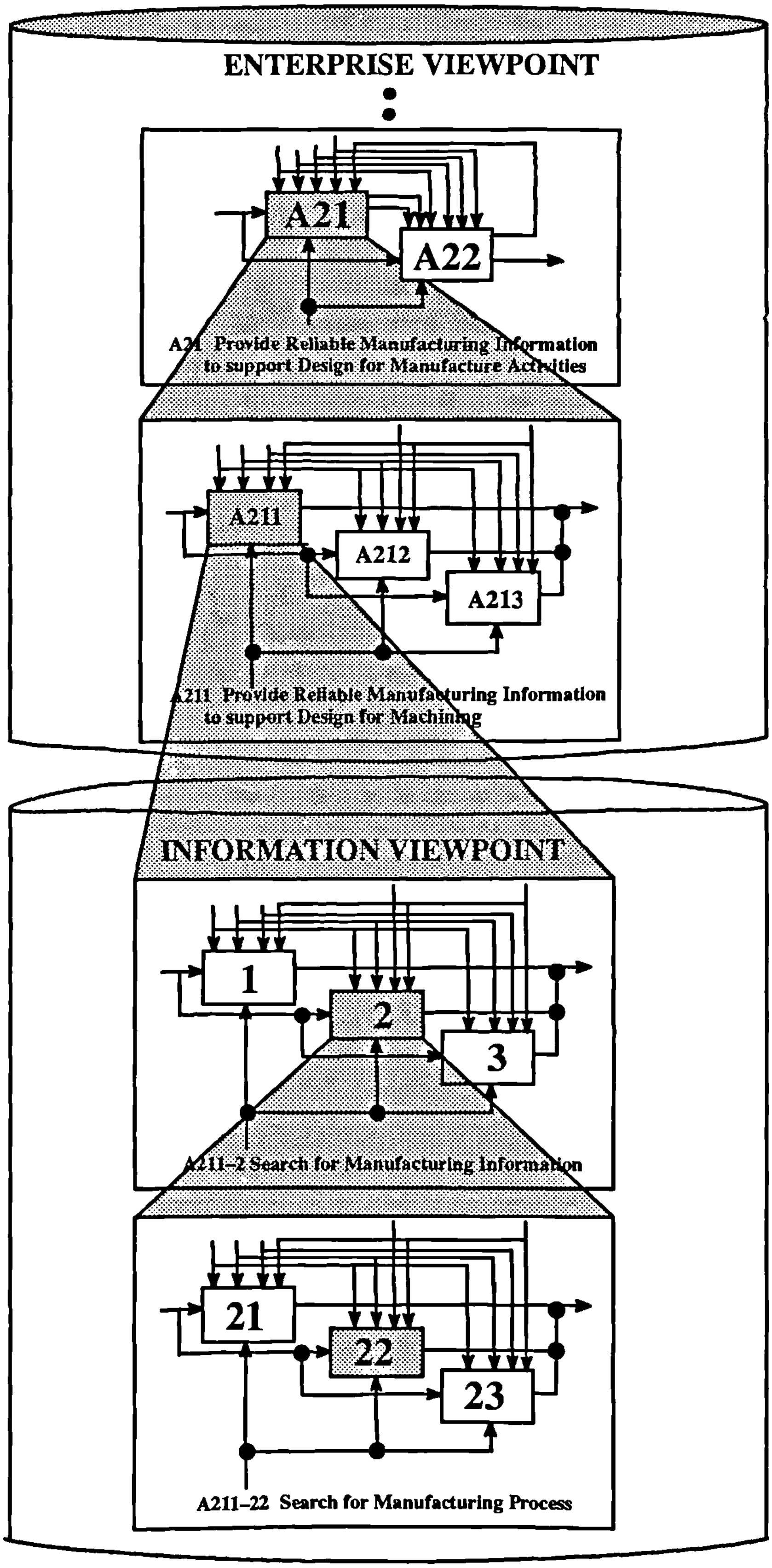
A. Molina	Figure 9.1 The MOSES CAE Reference Model based on RM–ODP	LUT–SE Research Group



A. Molina	Figure 9.2 The Contents of the MOSES CAE–RM related to the Manufacturing Model	LUT–SE Research Group



A. Molina	Figure 9.3 Relation between the MOSES CAE–RM Enterprise Viewpoint and the Manufacturing Model Enterprise Viewpoint	LUT–SE Research Group



A. Molina	Figure 9.4 Mapping Enterprise Viewpoint to Information Viewpoint in the Manufacturing Model	LUT–SE Research Group

Chapter 10

The Selection of the MOSES CAE Reference Model Methodologies and Tools

10.1. Introduction

This chapter describes the process, carried out by the author, to analyse and select the methodologies and tools which fulfil the requirements specified by the RM–ODP viewpoints in order to define the MOSES CAE Reference Model (CAE–RM). The requirements for each of RM–ODP viewpoint languages are described and explained. The reasons for chosing the IDEF0 method, EXPRESS language and Booch method so as to match those requirements are offered. A discussion about the advantages and disadvantages of using computer tools (RATIONAL ROSE and ST–DEVELOPER) to support the use of these methodologies is presented. Finally, the relevant characteristics of the Object Oriented Database (DEC Object/DB) and Object Oriented Language (C++) employed to develop the prototype of the Manufacturing Model are described.

10.2. Viewpoints and Languages Requirements

In this research the author studied thoroughly the three main documents of RM–ODP (ISO/IEC 10746–1 | ITU–T Rec. X.901, ISO/IEC 10746–2 | ITU–T Rec. X.902, and ISO/IEC 10746–3 | ITU–T Rec X.903) in order to understand better the requirements imposed by RM–ODP, and be able to define which methodologies and/or languages were the most appropriate to define the three RM–ODP viewpoints of relevance to this research i.e. Enterprise, Information and Computation (see Chapter 9, Section 9.2 for an explanation on the importance of these viewpoints for this research).

These documents define a structured set of concepts in terms of which a system can be represented from a particular viewpoint. This set of concepts provides a language for writing specifications of systems from that viewpoint, and such a specification constitutes a model of a system in terms of the concepts. Thus, for each of the viewpoints, a language has been defined for writing specifications of ODP systems. The terms of each viewpoint language, and the rules applying to the use of those terms, are defined using object modelling techniques. Each language is defined to be used for the specification of an ODP system from the corresponding viewpoint. These languages are called the enterprise language, information language, computational language, engineering language and technology language.

Each language uses concepts taken from ISO/IEC 10746–2, and introduces refinements of those concepts, prescriptive rules and/or additional, viewpoint specific, concepts relevant to the nature of specification concerned. The concepts and rules of the viewpoint languages are structured so that consistency constraints can be expressed between specifications using the languages.

The author noted, while studying the documents, that a viewpoint language is not defined in order to replace existing languages that are appropriate for that viewpoint; its purpose is to specify the set of concepts in terms of which specifications from that viewpoint must be structured in order to enable coordination and consistency with specifications from other viewpoints. Therefore, any existing language can, in principle, be used for specification of a system from a particular viewpoint provided that those specifications can be interpreted in terms of relevant viewpoint concepts.

10.3. The Mapped Methodologies to Viewpoints and Languages

The viewpoints of relevance for this thesis are: Enterprise, Information and Computation (Chapter 9, Section 9.2). Therefore only those viewpoints with their related language definitions are described in the next subsections. From the methodologies reviewed in Chapter 2 Section 2.4, the ones that best matched the concepts defined in the viewpoint languages were selected. These methods and languages constitutes the viewpoint definitions required to represent the MOSES CAE–RM in order to be able to describe the

CAE system and its elements (e.g. Manufacturing Model, Design for Manufacture Environment).

During the research, the author found that the Booch methodology could become the underlying methodology for the development of the CAE System, and therefore, the Manufacturing Model. The Booch method has three major phases:

1. Requirement Analysis – this involves the creation of a high level statement describing the purpose of the system and its scope.

2. Domain analysis – this determines the logical structure of the system.

3. Design – the physical structure of the system is determined and the logical structure is mapped to this. This leads to a working system prototype.

The figure 10.1 shows how a mapping between the Booch methodology, the MOSES CAE–RM and the concepts defined in RM–ODP is possible. The requirement analysis phase of the Booch Methodology assists in the definition of the Enterprise Viewpoint. The creation of the EXPRESS model was carried out by following the domain analysis process defined by the Booch Method. In this process the classes of objects, relationships among objects, attributes and constraints are defined using class diagrams, contains and uses relationships and inheritance diagrams. The identification of the key functions that the CAE system is to perform complements the information viewpoint definition and enables the overall understanding of the system functionality. This is represented by the IDEF0 model. Finally, the design phase and its related diagrams (class–category, design–class, design object) enable the description of the Computation Viewpoint.

10.3.1. Enterprise Viewpoint represented using IDEF0 Method

The purpose of the Enterprise Viewpoint is to enable specification of the system in order to serve the following objectives (ISO/IEC 10746–1):

- give the members of an enterprise an understanding of the purpose and scope of the system by showing correspondence between functions in the system and the enterprise requirements for that system;
- facilitate the specification of system functions in terms of their purpose and requirements and to link these to the other viewpoints.

These objectives were taken into account by the author to define the downward correspondence from the CIMOSA Requirements Definition Model to the Enterprise Viewpoint of the MOSES CAE–RM. This was possible because the Enterprise Viewpoint aims to show what functions the system provides, but not how these functions are implemented. These specifications define the boundary between the user who selects and utilizes the system and the designer who develops and implements it. At this point of the research, the author realized that the definition of the Enterprise Viewpoint of a system should not be confused with a total enterprise specification, i.e. this viewpoint is limited to the description of the system functions. By keeping this in mind, details about the enterprise itself, if required, can be modelled using the constructs defined in the CIMOSA architecture framework, therefore the CIMOSA Enterprise Requirements Model can be seen as an extension of the Enterprise Viewpoint. (See Chapter 8 for the detailed description of the CIMOSA model and its relation to the Enterprise Viewpoint).

The enterprise language defined in the RM–ODP documentation (ISO/IEC 10746–3) comprises concepts, rules and structures for the specification of a system from the enterprise viewpoint. The enterprise language defines basically three constructs: performative actions, agents and artifacts. Performative actions are activities which changes the state of objects. Agents are objects which initiate performative actions and artifacts are object which do not.

The author reviewed different methodologies for process modelling (e.g. IDEF0, SAMM, IDEF3), and two object oriented methods (Booch 1991, and Rumbaugh et al. 1991) to find out which method or language could better match the specification of the enterprise language. The IDEF0 method was though to be the most appropriate to capture these definitions. The IDEF0 method was chosen because is easy to use, easy to understand, widely used in engineering, and the MOSES research group agreed on it. Even when IDEF0 is a static model, and is not object oriented, the author believes that an IDEF0 representation is complete and consistent enough to represent the functionality provided by the CAE system, and therefore, the Manufacturing Model. The author matched the enterprise language definition in the following manner: performative actions were defined as activities in the IDEF0 model and, related inputs/outputs, controls and resources were either agents or artifacts (Figure 10.2).

The activities described in IDEF0 at the enterprise viewpoint represent the functions provided by the CAE system and its elements (e.g. Manufacturing Model, Design for Manufacture Environment, etc.). These functions (activities) can be derived further, within the same viewpoint as required, until the necessary level of detailed description is reached (Figure 10.3). Furthermore, correspondence will be possible with mapping the functions defined at the Enterprise Viewpoint onto the functions defined at the Information Viewpoint. This mapping allows a consistent functional representation of the MOSES CAE–RM. The relation between the Enterprise Viewpoint and the Information Viewpoint has been explained in Chapter 9, Section 9.4.

10.3.2. Information Viewpoint represented using IDEF0 Method and EXPRESS Language

The Information Viewpoint is represented in terms of information objects and their relationships, where information objects are abstractions of entities that occur in the real world, in the system, or in other viewpoints. To construct an Information Viewpoint of a system the following items must be considered (ISO/IEC 10746–1):

- the forms in which the information and information processing are visible to the users in the enterprise
- structures of information entities that define the information content of the enterprise
- rules stating the relationships between information entities

- quality attributes for information entities
- information flows
- the changes and derivations of information taking place as a result of information processing and the rules of these;

In order to be able to represent all these items at the Information Viewpoint, the RM–ODP specified the need to use two types of representation:

- an information activity model
- an information schema (information model)

The information activity model prescribes the activities, called information activities in RM–ODP, which are engaged by the system. In the case of the MOSES CAE–RM, the author decided to describe the activities which are carried out by the CAE system elements in order to achieve the CAE system functionality. By doing so, the Information Viewpoint represents how the functions of the CAE system are implemented.

The concepts defined in ISO/IEC 10746–3 to specify an information activity model can be straightforward mapped to the IDEF0, therefore the IDEF0 method was used to describe this model. The use of IDEF0 allowed the author to ensure model integrity, and consistency between the Enterprise and Information Viewpoint. The mapping from the IDEF0 at the Enterprise Viewpoint to the IDEF0 at the Information Viewpoint was direct and therefore the model integration was accomplished. The verification of consistency between viewpoints was quite simple, as the IDEF0 model at the Information Viewpoint was an extension of the model at the Enterprise Viewpoint. In Chapter 9, Section 9.3 the details of this mapping was discussed.

Together with the information activity model, an information specification has to be defined to comply with the RM–ODP specifications. An information specification is a schema which defines the classes of information entities, relationships between these information entities and possible actions, in some universe of discourse. In this PhD research this information specification is the Information Model. The information language defined in RM–ODP (ISO/IEC 10746–3, pages 8–10) specifies concepts such as schemas, integrity rules, cardinality constraints and relations which are also defined in the majority of the information modelling languages reviewed by the author (Chapter 2, Subsection 2.4.1). From these information modelling methods and languages (NIAM, EXPRESS, IDEF1x, Entity Relationship), EXPRESS (ISO DIS 10303–11) was found to be the more complete language.

The reasons of the author for chosing EXPRESS to represent the Information Viewpoint were:

- EXPRESS provides the mechanism for the description of information entities in a consistent manner. These entities are defined by using data elements, constraints, relationships, rules and functions.
- the information can be organized in a hierarchical way and can be structured in one or more schema which provides a basis for the partitioning and intercommunication of data.
- EXPRESS was designed so as to be both readable by humans and computer–interpretable by applications and supporting tools.
- it has a graphical and grammatical representation. Other information modelling methods such as NIAM only have a graphical representation.
- it is an international standard

It is important to mention that there are some concepts defined in the RM–ODP information language which are not defined in the EXPRESS language such as dynamic schema structuring rules. Nevertheless, extension of the EXPRESS language are being considered by ISO TC184/SC4 WG5 in order to enhances the specification capabilities of ISO 10303–11 to model the static and behavioural properties. This is very important because the models developed in this research by using EXPRESS could easily be modified to include those characteristics, if this is required in the future.

One of the most important reasons for documenting the information viewpoint using EXPRESS and IDEF0 models was to remain consistent with work carried out by certain groups within the International Standards Organisation, e.g. ISO/TC184/SC4/WG8. The work of this standardisation group has strongly influenced this Phd research.

An example of the Information Viewpoint of the Manufacturing Model is shown in figure 10.4. The activities that should be carried out by the Manufacturing Model to provide reliable manufacturing information are: search for manufacturing resources, search for manufacturing processes, and search for manufacturing strategies. The results of the search process is manufacturing information which has been captured and represented in the information schema defined in EXPRESS.

10.3.3. Computational Viewpoint represented using Booch Method

At the computational viewpoint the CAE system is viewed as a set of interacting components (objects) providing application specific functions supported by an infrastructure. The infrastructure is an environment for application functions created by the provision of a kernel and a set of common functions. The kernel provides for distribution transparent interaction between components implementing both application–specific and common functions.

The computational language comprises concepts, rules and structures for the specification of system, and all together constitutes the computational viewpoint. A computational specification defines, in distributions transparency terms (ISO/IEC 10746–3, pages 10):

- the components (objects) within a system;
- the activities that occur in a system
- interactions between components of a system
- structuring of components for inter–working and portability

The computation language comprises:

- interaction and types rules which give consistent distributed transparent access to all interfaces

- general activity rules that apply to all computation components (objects) in order to sustain the interaction and types rules

- portability rules for computational objects that are capable of implementation in terms of the distributed infrastructure defined by the engineering language

From a computational point of view, systems consist of configurations of computational objects. As the concepts defined in the RM–ODP computation viewpoint are based on object oriented modelling techniques, the author found the Booch Object Oriented Methodology to be appropriate to meet the viewpoint requirements. As a matter of fact, other object oriented methodologies were found, as well, to be suitable to represent this viewpoint. Nevertheless, the reasons for chosing the Booch Methodology (Booch 1991) among others (Wirfs–Brock et al. 1990, Coad and Yourdon 1990, Rumbaugh et al. 1991) were: previous experience on the use of the methodology, documentation readily available, Booch method is one of the leading emerging standards for object oriented analysis and design, and the availability of computer support tools. In fact, one tool (RATIONAL ROSE) was used in this research project (see next subsection 10.4).

The author's definition for the Computational Viewpoint of the MOSES CAE–RM is that this viewpoint defines all the CAE system components as objects with interfaces between them. There are different levels of abstraction at this viewpoint. At the highest level of abstraction the CAE system can be represented as the following set of objects: Information Models, Design for Function Environment, Design for Manufacture Environment, Manufacturing Information Generation Environment, and Engineering Moderator (Figure 10.5). Each of these objects has to be described in terms of its constraints, activities and interactions. The complete description of the Computation Viewpoint of the Manufacturing Model is presented in appendix J.

10.4. Computer Based Modelling Tools

Computer based tools assist the developers in the usage of modelling methodologies. In this research two computer tools, RATIONAL ROSE and ST–DEVELOPER, were used to facilitate the use of the Booch Object Oriented Methodology and EXPRESS language, respectively. These tools were proven to be very helpful for documenting and testing the models. In addition, these tools provided the means to standardise the models and improve their management and maintenance.

The major drawback of these computer support tools is their lack of integration. There is not automatic translation between Booch and EXPRESS. RATIONALE ROSE can generate C++ code, and so can ST–DEVELOPER. Nevertheless, the code generated by the two applications is generally different and a great effort is required to match code specifications. There is a need to integrate both tools, in order to be able to generate automatically an EXPRESS model from a Booch definition and vice–versa. The development of such tools is feasible, in the author's opinion, but a lot of time and effort is required to undertake such task.

10.4.1. Booch Modelling Tool: RATIONAL ROSE

The need for CASE (Computer Aided Software Engineering) tools to support the object oriented development has been recognised by several authors (Booch 1991, Coad and Yourdon 1990, Rumbaugh et al. 1991). These authors have defined as essential components of a toolset for object oriented development the following:

- a graphics–based system supporting the object–oriented notation for analysis and design.
- a browser which allows the navigation in the class structure and module architecture of a system
- tools for configuration management and version control
- a class librarian

These components should be present in any computer tool which aims to support an object oriented methodology. In the MOSES research project it was decided to use a CASE tool to support the use of the Booch methodology. The idea was to explore and assess the benefits of using such CASE tools during the development process of the CAE System elements, especially the Manufacturing Model, Design for Manufacture Environment and Engineering Moderator. The opportunity to use such a tool was made available due to the fact that Booch was used previously in the LUT–SE research group (Al–Ashaab 1994). Therefore the use of RATIONAL ROSE (Rational ROSE 1993) software to assist the researchers in the development of the Manufacturing Model, Design for Manufacture Environment and Engineering Moderator was proposed.

The RATIONAL ROSE software supports the use of the Booch methodology and enables the construction of complex object oriented systems. It automates the Booch notation and provides the four essential components defined above i.e. graphical editor, browser, tool manager and class libraries.

This tool was used in this PhD research to document and develop the Manufacturing Model. The Manufacturing Model is fully documented in Booch, and the description of the information entities (i.e. resources, processes and strategies), related taxonomies and relationships are captured in electronic form in this model. The Manufacturing Model supported by the RATIONAL ROSE software is a useful information model which can be used as a reference to search for the definitions and composition of the manufacturing resources, processes and strategies. This model is the basis for the EXPRESS model.

The use of this CASE tool in this project, in the author's opinion, has improved the communication of ideas and concepts among the researchers, and has promoted the good practice of keeping updated system documentation. As a matter of fact, it was found to be more easy to explain concepts and ideas using Booch notation than EXPRESS. A major problem with this tool, specific to this research project, is that there is no automatic translation from a Booch representation into an EXPRESS model. Therefore, once the Booch model was completed, the author translated it manually to EXPRESS. This makes the task of model maintenance a little bit more difficult, due to the fact, that if a change is made in the EXPRESS model, it has to be translated to Booch and vice–versa. Nevertheless, the author believes that this is a minor drawback and the benefits received from using RATIONAL ROSE have been greater.

10.4.2. STEP/EXPRESS Modelling Kit: ST–DEVELOPER

In the development of information models based on EXPRESS is important to have access to a set of utilities which facilitates their development. A set of utilities for EXPRESS should assist the developer in the following tasks:

- Compilation and Validation of EXPRESS schemas.
- Debugging of EXPRESS schemas through the use of editors and data browsers.
- Generation of useful code for Oriented Languages (e.g. C++) and Data Definition Languages for different Relational and Object Oriented Databases.
- Documentation of EXPRESS models (e.g. EXPRESS code formatting tool, EXPRESS–G).

An important characteristic of such sets of utilities is that they have to comply with the EXPRESS language defined in the ISO DIS 10303–11, and STEP standards parts 21 and 22 (ISO CD 10303–1).

Nowadays there are two type of utilities available for EXPRESS development: public domain software and commercial applications. The first type of software generally have been developed by universities or research centres to fulfil the requirements of research projects. Even though, these pieces of software are sometimes very useful, they do not have any support and usually a lot of bugs can be found while in use. An example of this type of utility is the NIST EXPRESS–Toolkit (Wilson 1993). The latter is software supported by a company which are committed to supporting and developing new and better EXPRESS tools. An example of this software is the ST–DEVELOPER, previously named STEP Programmer's Tool Kit, supported by STEP Tools Inc.

After reviewing the tools available in the public domain software the conclusion of not using them was drawn. Because they were not reliable and none of them provide a full set of tools required for this research project. A comparison study then was carried out to decide between two commercial application: STEP Programmer's Tool Kit and DECExpress. The study showed that the STEP Tool Kit was the best choice as the set of utilities provided by this product covered all the requirements of this research project i.e. EXPRESS compiler, EXPRESS–G generator and EXPRESS browser.

The ST–DEVELOPER, formerly called the STEP Programmer's Tool Kit, contains a set of utilities very useful for the development of EXPRESS information models, such as: EXPRESS–G tool set, EXPRESS compiler, EXPRESS interpreter, programming libraries, SDAI and ROSE Libraries, and STEP physical file support. In addition the platforms where this system can run cover a wide range of UNIX systems and PC/Microsoft Windows.

The author's experience of the use of the ST–DEVELOPER was positive as the use of this software facilitated and improved the development of the EXPRESS models defined in the Manufacturing Model. In addition, the facilities provided by this tool simplified the hard and tedious work of model documentation i.e. EXPRESS–G models were generated easily. A disadvantage of the ST–DEVELOPER is that it does not produce code for the Data Definition Language of the Object Oriented Database employed in this research. A software prototype to perform this translation exists but it was not used because it was not is available. Therefore the translation between the EXPRESS model and the Data Definition Language of the Object Oriented Database was carried out by the author manually.

10.5. Object Oriented Database and Object Oriented Language

A prototype version of the Manufacturing Model has been implemented using the Object Oriented Database DEC Object/DB (Objectivity/DB 1991) and the Object Oriented Language C++ (Stroustrup 1986). This prototype is described in appendix J. The reason for using this software tools was that it was readily available from one of the collaborators in the MOSES research project.

The characteristics of DEC Object/DB allow the most straight forward implementation from our Booch and EXPRESS definitions. In a traditional relational database there are often a small number of entity types with a large number of instances of each type. Often only simple, fixed relationships exist between entity types. In the Manufacturing Model there are many entity types with fewer instances of each type. Complex relationships exist between entity types, and new relationships may be created e.g. a machine tool using a new cutter type may be able to perform a variety of additional processes. The ability to create a number of objects, and make complex associations between them, enables us to more realistically represent the capabilities of resources, the relationships between resources and to map from process requirements to suitable resources.

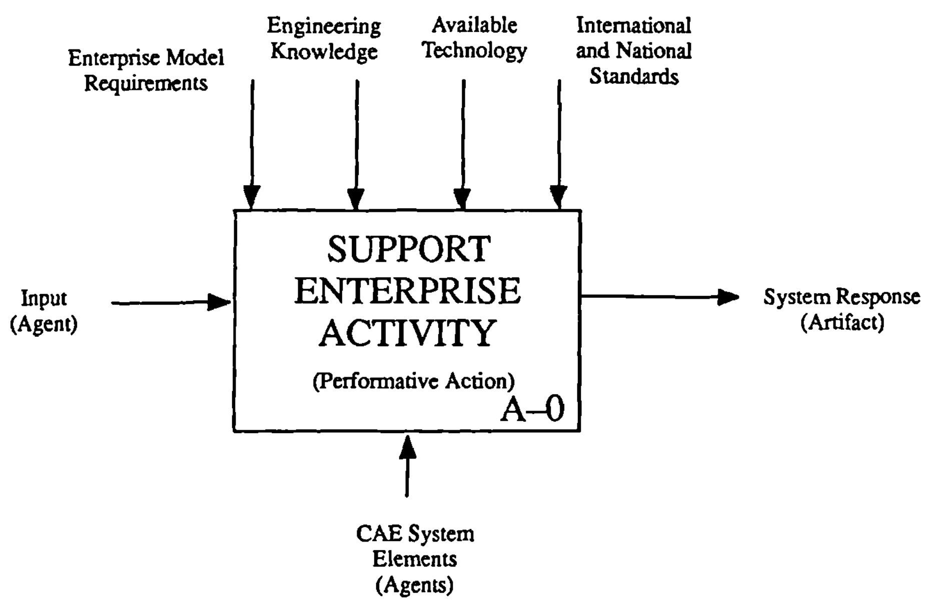
The manufacturing environment may change with the addition or removal of resources or process capabilities and so this must be reflected in the contents of the Manufacturing Model. A system that supports concurrent engineering will need to deal with versions of data items. Dependence on versions, and relationships between versions must be explicitly recorded. DEC Object /DB has facilities for monitoring versions of objects. This element of system development will need to be explored in greater depth when the future behaviour element of the Manufacturing Model is more fully developed.

Usually a database system has the advantage of providing a general language for data definition and manipulation. The DEC Object/DB system has extended the C++ programming language with appropriate persistent data structuring facilities to provide query language facilities. Whilst the system is powerful, a high degree of programming expertise is required to build applications for the management and manipulation of data within the database. The development of commercial systems would be greatly eased if a more high level query language existed. The advantage of the extended programming language approach is that it can facilitate a more seamless integration between application programs and the database system.

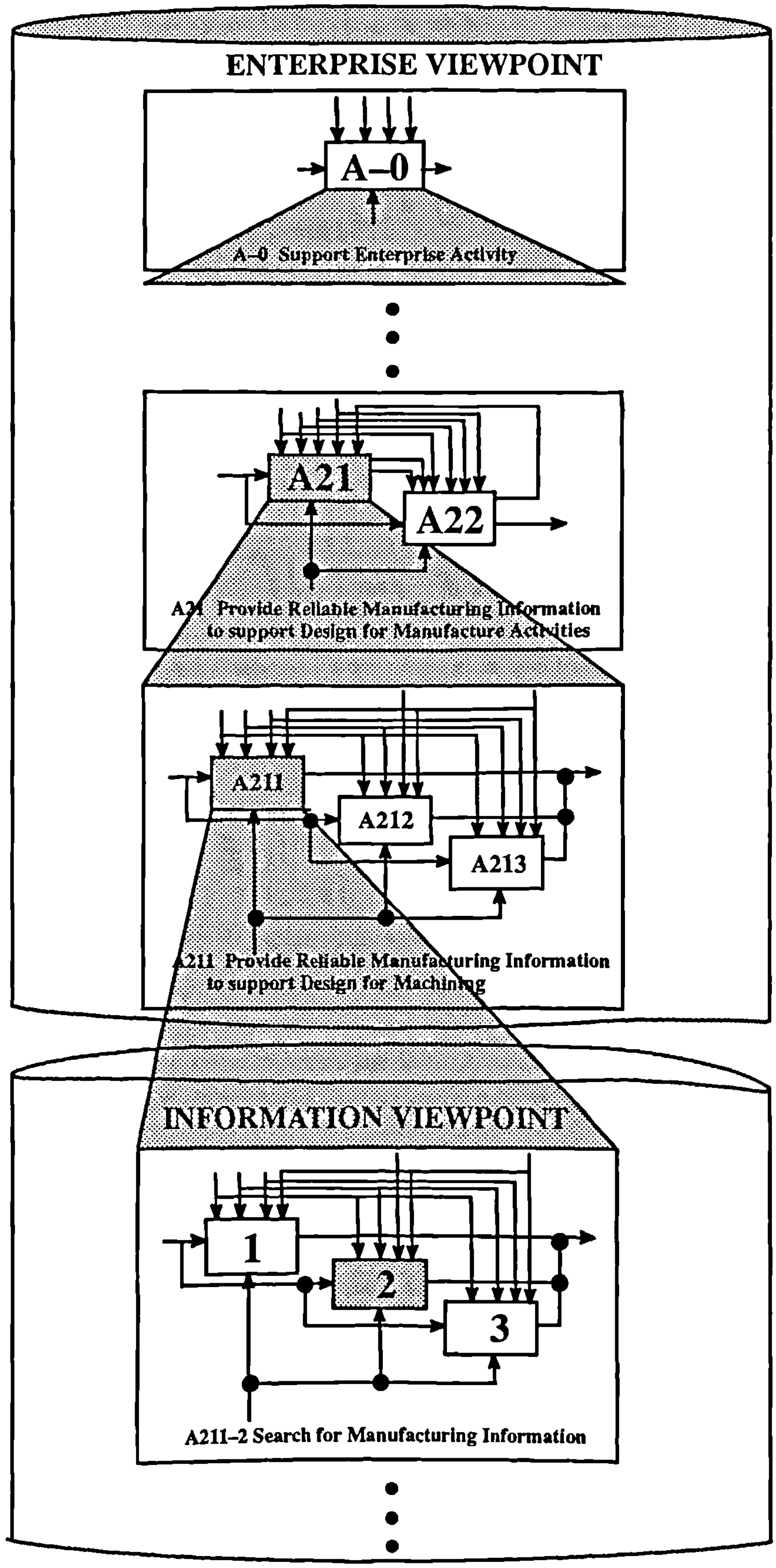
In general, DEC Object/DB and C++ are very useful software tools. The flexibility and capabilities of such tools enabled the author to implement a prototype version of the Manufacturing Model, with enough functionality to prove the concepts of this thesis in a time span considerable less than if traditional tools had been used (e.g. Relational Database, C programming). However it was required to translate manually the EXPRESS model into the Data Definition Language of Objectivity as there are not effective automated tools to do this task. Therefore some time was spent to find out what was the best way to implement the EXPRESS model in the Objectivity Database.

BOOCH PHASES	BOOCH DESCRIPTION	CAE–RM DESCRIPTION	RM–ODP DEFINITION
Requirements Analysis	System Function Statement	IDEF0 Control, Output, Function, Input, Mechanism	Enterprise Viewpoint: Specification of Requirements for and policy statements about, ODP systems
Domain Analysis	Key–abstraction Diagram Inheritance Diagrams Object–scenario Diagrams Class Specifications Object Specifications	IDEF0 and EXPRESS EXPRESS Schema Type Entity	Information Viewpoint: Expression of information and information processing functions in ODP systems
System Design	Class–category Diagram Design–class Diagram Design–object Diagram Specifications Architecture Description Prototype Plans	BOOCH System Structure Model DFF, MIG, IM, EM, DFM	Computation Viewpoint: Expression of the functional decomposition of an ODP system, and of the interworking and portability of ODP functions.

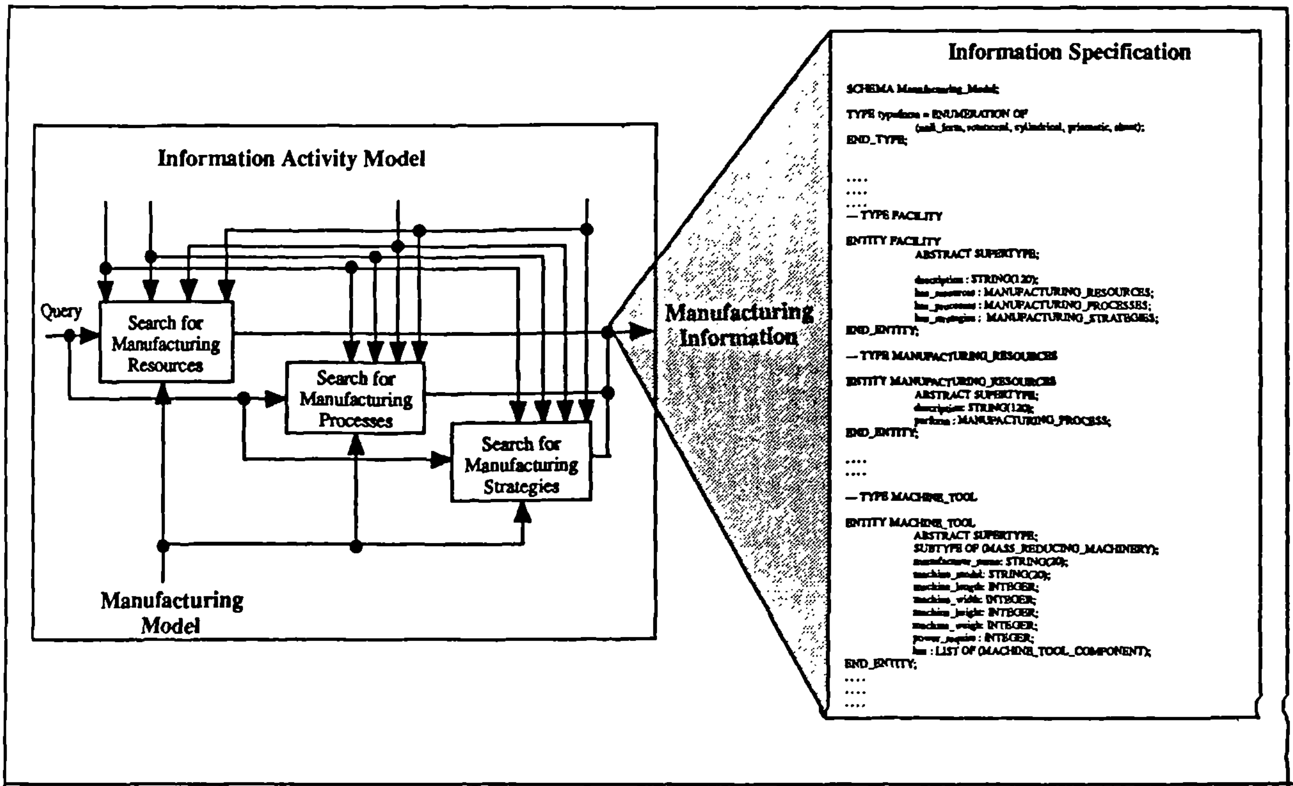
A. Molina	Figure 10.1 Booch Methodology, MOSES CAE–RM and RM–ODP Descriptions and Relations	LUT–SE Research Group



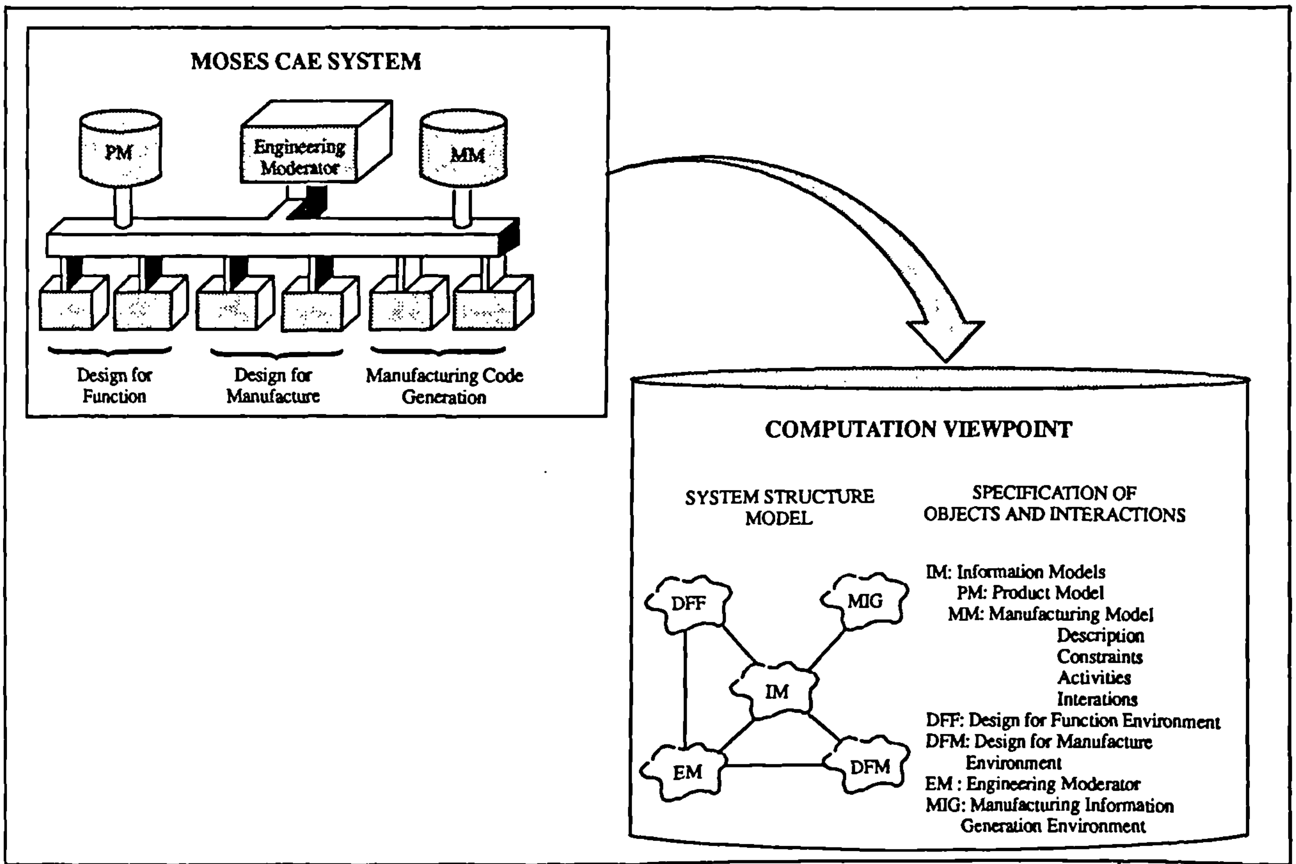
A. Molina	Figure 10.2 Mapping the Enterprise Viewpoint Concepts to IDEF0 Model	LUT–SE Research Group



A. Molina	Figure 10.3 Mapping Enterprise Viewpoint to Information Viewpoint	LUT–SE Research Group



A. Molina	Figure 10.4 Information Viewpoint represented with IDEF0 and EXPRESS Language	LUT–SE Research Group



A. Molina	Figure 10.5 Computation Viewpoint represented with Booch Notation	LUT–SE Research Group

Chapter 11

The Specification of a Manufacturing Information Modelling Methodology

11.1. Introduction

This chapter describes the integrated methodology conceived by the author to be used in the information modelling work. The modelling dimensions (i.e. data, function and behaviour) considered necessary to represent the manufacturing capabilities are explained. The role of manufacturing resources, processes and strategies to represent generic and company specific manufacturing information is outlined. The need to represent strategic and operational information is discussed and the multi–level structure (i.e. Factory, Shop, Cell and Station) defined by the author is described. Finally, examples of the use of the manufacturing information modelling methodology are presented.

11.2. The Manufacturing Information Modelling Methodology

In the chapter 8, section 8.3, the author explained how the CIMOSA Requirements Definition Model allows the identification of the requirements for manufacturing information. This information has to be modelled in order to capture and represent the most relevant aspect of a manufacturing facility in this research i.e. the manufacturing capability. The Information View of the MOSES CAE–RM aims to represent the information models required to support the design and manufacturing activities.

In the MOSES CAE–RM (Chapter 9, Subsection 9.3), it was decided to describe the Information Viewpoint by using the IDEF0 methodology and EXPRESS language. Therefore the main objective of the author at this stage of the research was to define and create the Information Viewpoint of the MOSES CAE–RM related to the Manufacturing Model, where its functionality (information activity model) could be described and documented using IDEF0 and its information schema (information model) could be represented using the information modelling language EXPRESS, see figure 11.1

The Manufacturing Model research is mainly concerned with the representation of the manufacturing capability information of a facility. In order to facilitate this task, the author has defined a Manufacturing Information Modelling Methodology to be used to perform the modelling of manufacturing information in the LUT MOSES research group. The methodology was defined to:

1. set a context and harmonize the research in modelling the manufacturing information
2. establish a formal method for the development of the Manufacturing Model.
3. build a consistent, complete and sound Manufacturing Model.
4. structure and couple other research contributions related to the Manufacturing Model research, such as Manufacturing Strategist (Ellis 1995) and Injection Moulding (Lee 1993, Al–Ashaab 1994).

The idea of using a multi–dimensional representation to describe the integrated methodology was taken from CIMOSA. The author recognised the need to represent all the elements involved in the Manufacturing Model research in a complete frame illustrated in the figure 11.2. This multi–dimensional representation contains all the elements that the author considers to be necessary to adequately describe the manufacturing capability of facility. The methodology is composed of:

1. The modelling dimensions: data, function and behaviour
2. The manufacturing information entities: resources, processes and strategies.
3. The organization of these manufacturing information entities into a multi–level structure: Factory, Shop, Cell and Station

This multi–dimensional representation helped the author to identify the following issues about the Manufacturing Model research:

- The need to model different aspects of the manufacturing resources, processes and strategies to capture their relevant characteristics in order to support strategic and operational decisions. These aspects are called in this thesis modelling dimensions and allow the representation of: data (class, composition and attributes), function (functional capabilities), and behaviour (status and performance).

- The importance to capture the manufacturing facility capability using generic and company specific manufacturing information. The manufacturing resources and processes describe generic manufacturing information. Similar types of manufacturing resources and processes can be used by different companies. In fact, two companies can have the same type of technology, i.e. manufacturing resources and processes. Nevertheless, the manufacturing facility of each of these companies could perform in a different way because of the company's decisions on how to organise and use those resources and processes. This aspect must be captured in order to truly represent the manufacturing capability of a company. The manufacturing strategies represent this company specific information which allows a company to specify how the resources and processes are organised and used in order to support the company's manufacturing function.

- The relevance of defining a multi–level structure to represent strategic and operational manufacturing information. This multi–level structure allows the representation at the Factory level of the strategic manufacturing information which defines what the company's manufacturing strategy is, and at the three lower levels, i.e. Shop, Cell and Station, the representation of the operational information which describes how the manufacturing strategy has been realized. The Factory level is aimed to represent the strategic information required for the formulation of the manufacturing strategy. The remainder three levels represent the operational information of the manufacturing facility to support product's life cycle activities (e.g. design and manufacture).

The following sections describes in detail how these three issues have been tackled by the author in order to define and develop a consistent and complete Manufacturing Model.

11.3. The need for different modelling dimensions

The need to identify the different modelling dimensions required to represent real objects that exist in an universe of discourse was one of the first challenges faced by the author. Based on past research experience (Molina et al. 1992), other research projects (CIMOSA, IMPPACT) and the modelling literature reviewed, especially Object Oriented Methods (Booch 1991, Wirfs–Brock et al. 1990, Coad and Yourdon 1990, Rumbaugh et al. 1991), the author realized that different properties of an object in the real world (universe of discourse) have to be modelled. These properties represent different aspects of an object which allow to capture the essence of that object. Questions which arise to the author were: What an object is and what is its role ? What are the relations between objects? What an object can do ? How the object evolves in time? How the object performs its task ? In order to be able to answer these questions, and therefore model the information regarding an object, the following modelling dimensions were identified by the author:

1. Modelling the data which describes what the object is and what it is its role in the universe of discourse.

2. Modelling the function to represent the relations between objects and what an object is able to do.

3. Modelling the behaviour to express how the object evolves in time and what is the object performance.

In order to explain what are these modelling dimensions in the Manufacturing Model, the author uses the graphical representation of conceptual graphs (Sowa 1984). The conceptual graphs allow the description of concepts, their attributes and relations by interconnecting them using static semantics, know as conceptual relations. In the graphical representation of Conceptual Graphs a rectangle represents a concept and a circle a conceptual relation. Using this simple representation the different aspects that each modelling dimension represents about the manufacturing entities (i.e. resources, processes and strategies) are explained in the following subsections.

11.3.1. Data Modelling

The author has used different modelling abstraction mechanisms employed by object oriented modelling techniques (Booch 1991, Rumbaugh et al. 1991) in order to describe what an object is and what is its role in an universe of discourse. In this research the objects are the manufacturing information entities (resources, processes and strategies) and the universe of discourse is the manufacturing facility. The modelling of the data about manufacturing entities requires:

- to capture the manufacturing entities in suitable abstractions. An abstraction denotes the essential and relevant characteristics of an entity. This is important in order to define clearly what information is required to be represented about the manufacturing resources, processes and strategies in order to support design and manufacturing activities.

- to develop adequate classifications or taxonomies of manufacturing entities in order to simplify the definition of manufacturing entities, and group them into classes which share characteristics and functionalities. This is very important in order to understand better the complex composition of manufacturing resources, processes and strategies and be able to define adequate abstractions.

- to define manufacturing entities based on the aggregation of their functional components. This permits a definition of manufacturing facilities by using simple characterisations of manufacturing resources, processes and strategies.

There are two abstraction mechanisms defined in object oriented modelling methods, which enabled the author to accomplish the above: generalisation/specialisation (*is–a*) and aggregation (*has*).

The generalisation/specialisation (*is–a* relation) simplifies the definition of manufacturing entities by generalising or specialising them regarding their type or class. Therefore manufacturing entities of the same type can share common characteristics. For example (Figure 11.3):

- a Machine Tool *is–a* Production Machinery, generalising Production Machinery *is–a* Production Resource, or specialising a Turning Centre *is–a* Machine Tool
- a Turning Process *is–a* Machining Process, in the same manner Machining Process *is–a* Mass Reducing Process, and Rough Turning *is–a* Turning Process
- an Operational Strategy *is–a* Manufacturing Strategy, as Manufacturing Strategy is a base definition it can not be generalised further in this research context. But different types or Operational Strategies exist,i.e. Control/Process Planning Strategy *is–a* Operational Strategy.

The use of this abstraction mechanism facilitates the organization of manufacturing entities in classifications (taxonomies of classes) in order to simplify their description. General attributes are specified at the top classes while specialized attributes are defined in the more specific classes. Because of the property of inheritance the more specialised classes will have the same attributes of their parent classes in addition to the ones defined for that particular class . This is one of the most powerful abstraction mechanism of object oriented methods (Booch 1991).

The aggregation (*has* relation) permits to compose manufacturing entities based on their functional components. The *has* relation is very important because it allows the complete definition and composition of resources, processes and strategies (Figure 11.3):

- a Machine Tool *has* Turret or a Machine Tool *has* Spindle, which are important components of a machine because they describe physical capabilities of a machine (e.g movements). A Machine Tool *has* attributes (identification number, size, weight, cost, etc.) which can be used for administrative and operational purposes.

- a Turning Process *has* Input and Outputs, where the Inputs and Outputs can specify material, information or resources. On the other hand, the Turning Process *has* attributes (identification, description, cost, times, etc.) which defines their capabilities.

- an Operational Strategy *has* objectives, decisions and rules/constraints. The objectives in a strategy are achieved by a set of decisions which can be enforced by applying rules which constrains the use of the resources and processes.

These abstraction mechanisms allow a classification of manufacturing process and resources to be made according to the ISO TC184/SC4/WG8. Even when this classification has been made for resources only, the author has extended it to represent manufacturing processes as well. The classification groups manufacturing resources, processes and strategies into:

- resource/process/strategies group: comprises resources/processes/strategies that are determined by a set of identical characteristics. A group hierarchy can be built following different hierarchy criteria.

- generic resource/process/strategies: is a resource/process/strategy class where the characteristics, parameters and assigned composition are completely defined.

- specific resource/process/strategies: is the instance of a generic resource/process/strategy where all the parameters have been instantiated, except temporal characteristics, such as material planing dates, wear conditions, state, status, etc.

- individual resource/process/strategies: is a concrete resource/process/strategy where individual and temporal characteristics (like material planing dates, wear conditions, state, status, etc.) can be assigned to.

The use of these abstraction mechanisms and classification based on ISO TC184/SC4/WG8 are explained in detail in the next chapter (Chapter 12, Section 12.3), the author believes that their usage in the examples presented in Chapter 12 will facilitate the reader to better understand these concepts.

11.3.2. Function Modelling

The description of system's functions through the process of functional decomposition and categorisation of the relations between functions is provided by modelling the function (Klein and Scheer 1990). The author's definition of modelling the function is related to the representation of the functional capabilities of a manufacturing entity i.e what a manufacturing resource, process or strategy is able to do.

In the Manufacturing Model the functional capabilities of the manufacturing entities are defined based on semantic relationships between objects. The characteristic of object oriented methods to represent semantic relations simplifies this task. For example, in figure 11.4 the following relationships between the manufacturing entities are defined in order to represent the functionality of resources, processes and strategies:

- a Turret *holds* Tools or the same machine tool *performs* (*perf*) Turning Process.
- a Turning Process *uses* Tools and Machine Tool, and the Turning Process capability is defined (*def*) based on the movement of the Tools and Workpiece
- a Operational Strategy *controls* the usage of resources, processes or facilities.

These semantic relations (*holds, uses, perf, def, and controls*), among others, facilitates the representation of the capabilities of the manufacturing entities. In figure 11.4, for example, the Machine Tool is able to perform certain processes (Turning Process) or able to use certain tools (Single Point Tools). Following the same example, the Turning Process is defined in terms of the movements of the tool and the workpiece, this definition describe the Turning Process and hence the capabilities of the turning process in terms of shapes it can produce based on its freedom of movements. On the other hand, because strategies represents how resources and process are organized the function dimension describes how an strategy controls a facility (e.g. Shop, Cell, Station), or its resources and processes. The complete set of semantic relations used in this research to describe capabilities are presented in Chapter 12 and Sections 12.4, 12.5 and 12.6.

11.3.3. Behaviour Modelling

Behaviour modelling describes the dynamics of a system. i.e. what is happening through time (Kinstrey et al. 1990). All entities in a system have some state in time, execute an operation, and exhibit some performance. Modelling the behaviour allows the representation of time dependencies (sequence, parallelism, concurrency), operation execution and performance. Modelling methods are under development, however there is not a known methodology that adequately describes dynamic behaviour.

The modelling of the behaviour is outside the scope of this research, but the author has defined it within the methodology for possible future research work which will need this modelling dimension (e.g simulation, factory modelling, real time control) and to have a complete and consistent methodology. In this research the dimension of behaviour represents a very limited description of how a manufacturing entity is executing and performing an action, and in what state and status the entity is. This dimension is described using semantic relations. The figure 11.5 illustrates the behaviour modelling:

- a Machine Tool can be at different states according to predefined conditions, for example, the Machine Tool can be at any of the following *states*: Halt, Loading–Work, Active or Unloading–Work.
- a Turning Process can be available or not (*status*), or it can be performed in–house or it can be subcontracted (*location – loc*).
- a special case of behaviour is described in the manufacturing strategies. The behaviour for the strategies represents the performance measures because these measures indicate how a strategy is performing (*perfor_measures*).

The reason for not fully exploring this modelling dimension, is due to the fact that the scope of the Manufacturing Model was limited to support concurrent design of products i.e. to support the Manufacturing Strategist and the Design for Machining applications

MOSES research project (Chapter 5, Section 5.3 – 5.4, Figure 5.2). Therefore, the author considered that this limited representation of behaviour was sufficient to achieve the objectives of this research i.e. represent the manufacturing capability.

11.4. The Description of Generic and Company's Specific Manufacturing Information

The author has identified two types of manufacturing capability information within a company. Information about the generic capability of the facility based on technical factors, and company's specific capability information based on organisational and operational factors.

The generic capability information is related to the technological aspects of a facility, i.e. the type of resources and process that the facility has. This information represents theoretical capabilities based on what the manufacturing resources and process are able to do, and therefore what the facility can produce. This information is related to the physical limitations of the resources and process which includes among others: product size, product weight, product form, volume, materials, tolerance, and surface finish. These capabilities are represented based on the technology employed by the resources and processes (Figure 11.6)

The second type of manufacturing capability is related to the strategic decisions and operational rules imposed on the use, organization and composition of the manufacturing resources and processes. The theoretical capability of a resource or process can be diminished depending on the strategies imposed on them. For example, figure 11.6 describes an operational strategy which restricts the use of a Machine Tool if the dimensions of the workpiece are less than 10 or greater than 100. This strategy reduces the theoretical capability of the Machine Tool but in general it is expected to improve the performance of the Machine Tool by reducing the set–up time. Therefore a strategy can represent restrictions or special uses of resources, process and facilities but aiming to improve their performance in order to achieve the manufacturing objectives of the company.

The inclusion of the strategies in the Manufacturing Model is a novel approach which enables the model to represent the company's specific capability information. Therefore the manufacturing resources and processes represent theoretical capability (generic capability information) and the strategies imposed on resources and processes, actual capability (company capability information).

11.5. The Representation of Strategic and Operational Information

Nowadays, manufacturing enterprises in general have adopted five or four functional levels to organise their function. A de–facto standard has already emerged, by the petition of various standardization bodies such as ISO and NIST (National Institute of Standards and Technology), together with various European projects within ESPRIT (BSI PD 6526:1990). This de–facto standard had partitioned the factory into five hierarchical levels: Facility (Factory), Shop (Area), Cell, Work station and Equipment.

This five hierarchical model has been used as a reference to structure the Manufacturing Model into the following levels (Figure 11.7):

1. Factory Level
2. Shop Level
3. Cell Level
4. Station Level

where the fifth level (Equipment) is enclosed at the Station Level. These levels of abstraction provide manufacturing information for all functional activities within a manufacturing enterprise. The different levels of the Manufacturing Model represent the different perspectives that users may have about the manufacturing information capability. These different perspectives represent different views about the resources, processes and the facilities, i.e. stations, cells and shops. As a matter of fact, this hierarchical organisation enables the Manufacturing Model to provide the adequate strategic and operational manufacturing information

The Factory level will hold strategic information which represents the manufacturing strategy of a company, the remainder three levels (Shop, Cell and Station) represent operational information which describes how the manufacturing strategy has been realized (Figure 11.7).

11.5.1. Factory Level

This level holds the strategic information, mainly concerned with the definition of strategic resources and strategic decisions which made the company's Manufacturing Strategy. The Manufacturing Strategy is the set of decisions which determines the capabilities of the manufacturing facility. These decisions influences the way the facility is structured, composed and organised (Figure 11.7). The three lower levels of the Manufacturing Model are, therefore, the representation of how these decisions have been realized.

The Factory Level describes the manufacturing strategy in terms of strategic decisions, operational rules and performance measures. The operational rules define how the strategic decisions are reinforced and achieved. The performance measures are defined in order to allow the evaluation of the manufacturing situation using the lower levels of the Manufacturing Model. By doing so, this level provides the information to assess the realization of a Manufacturing Strategy. The complete explanation about the representation of the manufacturing strategies in the Manufacturing Model is presented in Chapter 12, Section 12.6.

11.5.2. Shop, Cell and Station Level

Operational information regarding the composition, structure and organisation of the shops, cells and stations is represented at the three lower levels of the Manufacturing Model (Figure 11.7). The information represented at these levels is regarding the capability of the shop, cell or station, resource and process technology, composition of stations into cells and cells into shops, and the operational strategies imposed on the use of resources, process and facilities. A facility in this context can be a shop, a cell or a station. This information represents how the Manufacturing Strategy has been realized by putting together a set of manufacturing capabilities that enables a company to pursue its chosen competitive strategy over the long term.

The capability information for the different perspectives (i.e. managers, production engineers, manufacturing and design engineers) is defined at the appropriate level. For example, from the management point of view the relevant capability information is captured at the Shop level, i.e. processing cycle, cost, volume, etc. In the same manner, at the Station level the information regarding a specific machine tool and the different type of tools that can be used to perform certain processes is captured in order to support the manufacturing or design engineers. These representations have been very helpful in order to define complex facilities. The modelling of facilities, i.e. shops, cells and stations is presented in Chapter 12, Section 12.7.

11.6. The use of the Manufacturing Information Modelling Methodology

The main objective in modelling the manufacturing information is to describe and represent the manufacturing entities at different functional levels in terms of its data, function and behaviour. The following text describes an example of how these aspects can be modelled:

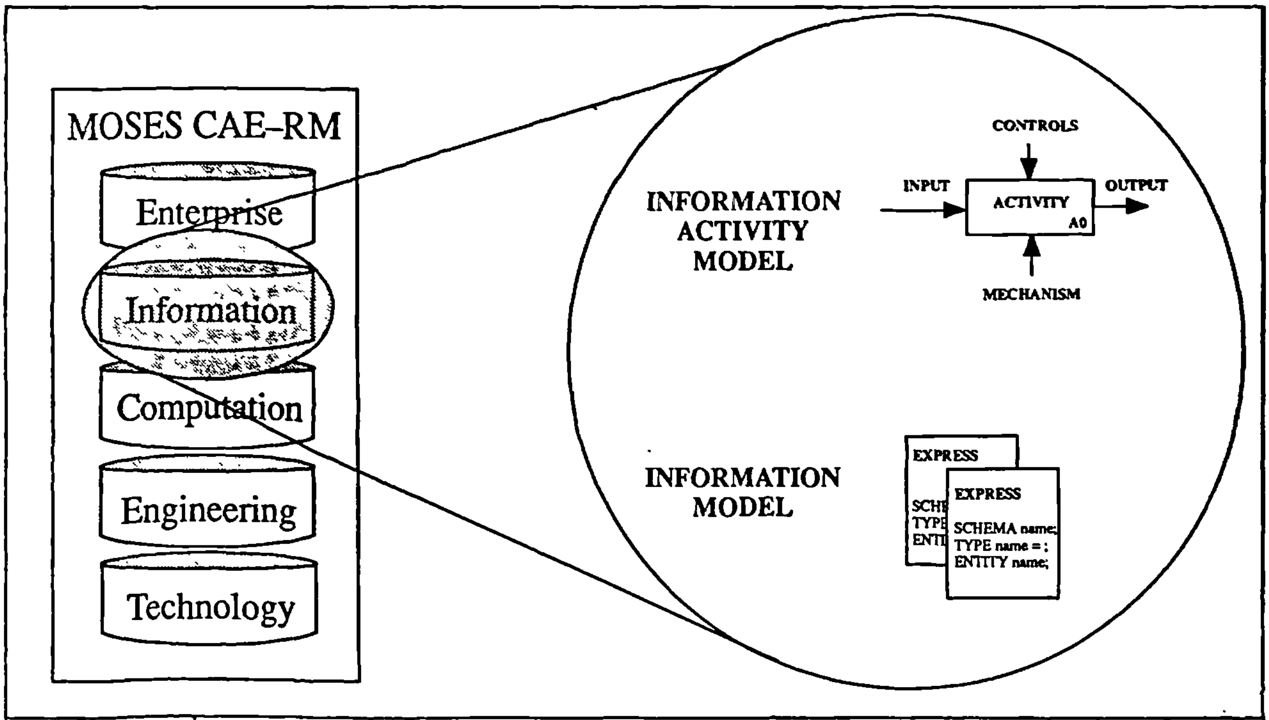
1. Modelling the Data: the representation of the data is carried out by defining taxonomies of resources, processes and strategies. In the example shown in figure 11.8, a simplify taxonomy of resources is presented. Then, the description of a particular resource, process or strategy based in the aggregation relation *has*, has to be modelled. In figure 11.8 a Turning Centre based on its components (turret and spindle) and attributes (administrative and technical) has been created. This

modelling activity has been carried out at the Station Level and aims to represent the capabilities of a Turning Centre.

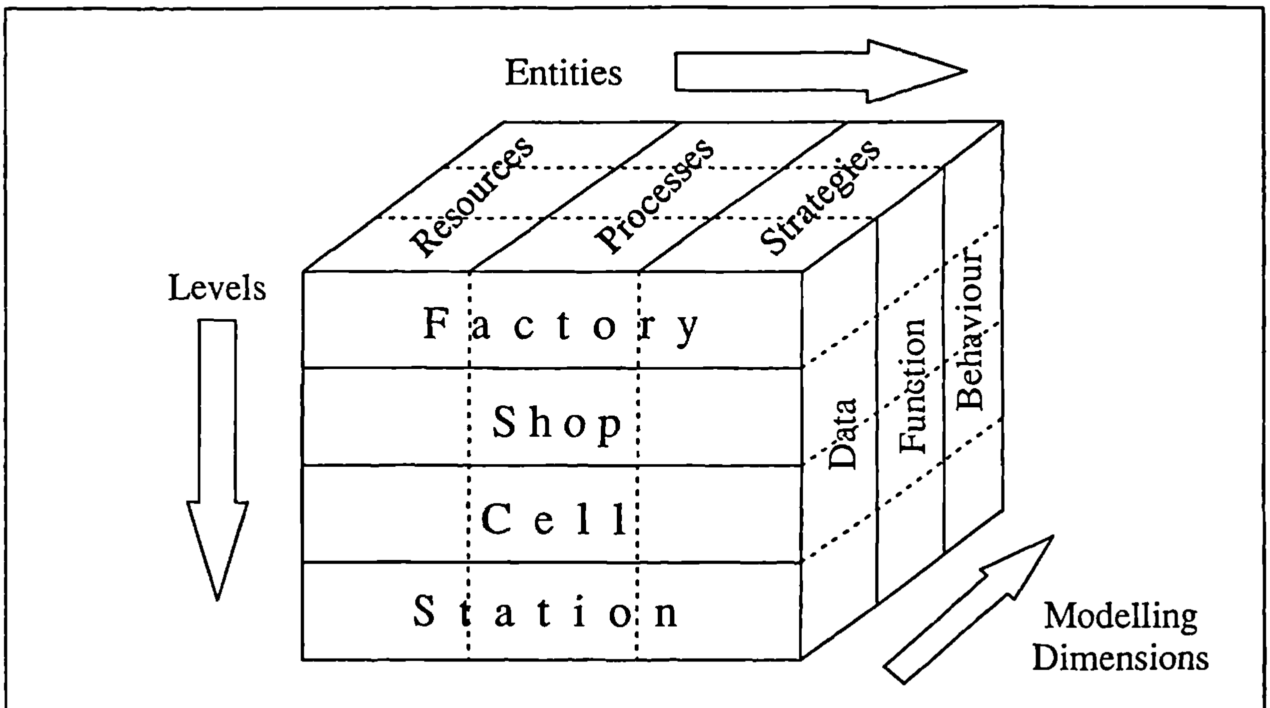
2. Modelling the Function: the semantic relations (*perf, uses, def, controls*) are used to link the entities required to represent the capability of a process or resource. In the figure 11.9 the capability of a Turning Centre to perform a Turning Process by using certain type of tools has been modelled. In a similar manner the Turning Process is related to the resources it uses, i.e. Turning Centre and Tools. Finally, the constrains and rules imposed on the Turning Centre and Turning Processes are defined by using their linkage to the operational strategies. In this case, constraints which limit the size and weight of components to be produced in the Turning Centre and the rule controlling the Turning Process of using only tools type 50.

3. Modelling the Behaviour: this aspect is modelled by using the semantic relations (*state, status and perfor–measures*). These relations allows the definition of situations about the resources and processes, and the identification of the important parameter to be evaluated to measure their performance. In the example illustrated in figure 11.10, the Turning Centre is active which means that the machine is in used, the Turning Process is available meaning that the manufacturing facility is able to perform Turning Processes and, finally, the important performance measures of the Turning Centre have been defined to be set–up time, machining cycle and percentage of scrap.

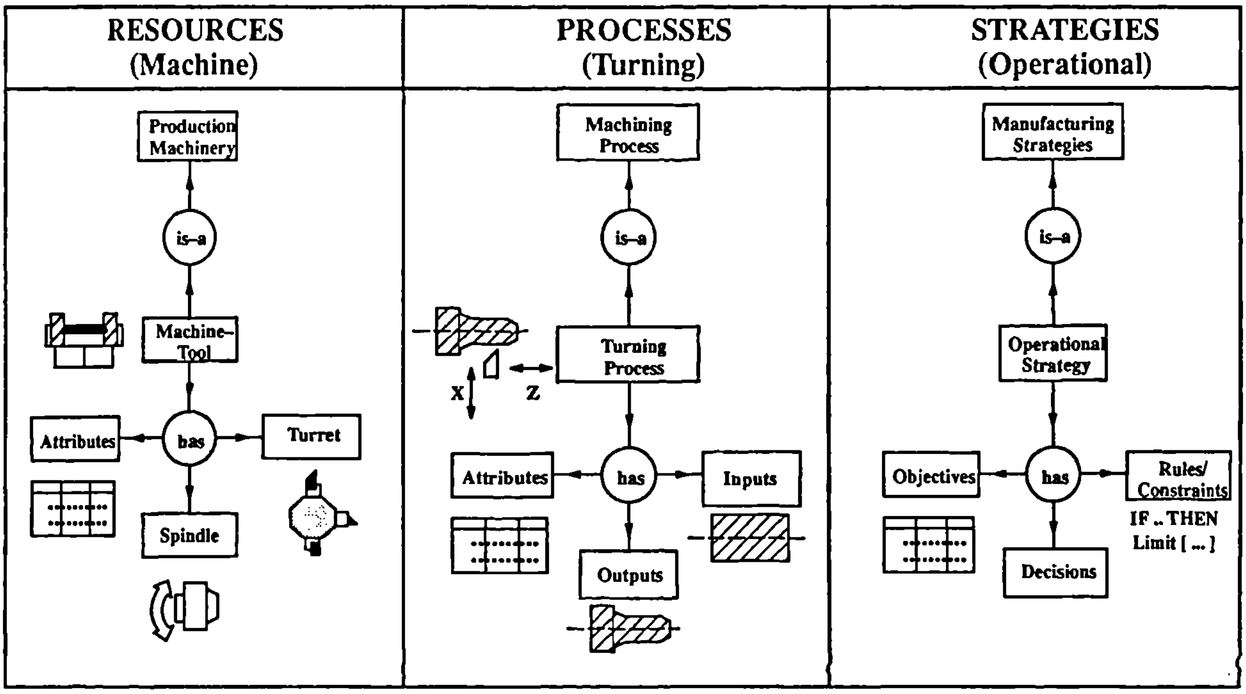
The modelling methodology defined above allowed the author to have a systematic approach to capture and represent the manufacturing information, and therefore build a consistent Manufacturing Model. In the next chapter the detailed aspects of the modelling work are described.



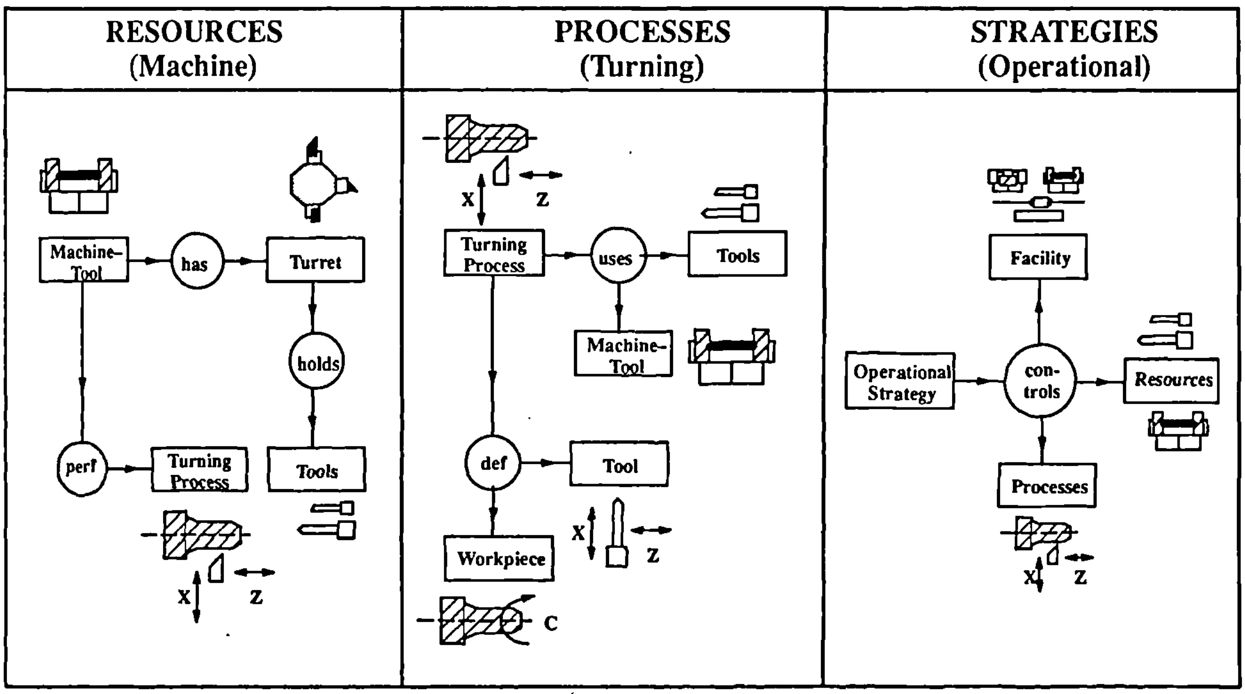
A. Molina	Figure 11.1 The Representation of the Information Viewpoint in IDEF0 and EXPRESS	LUT–SE Research Group



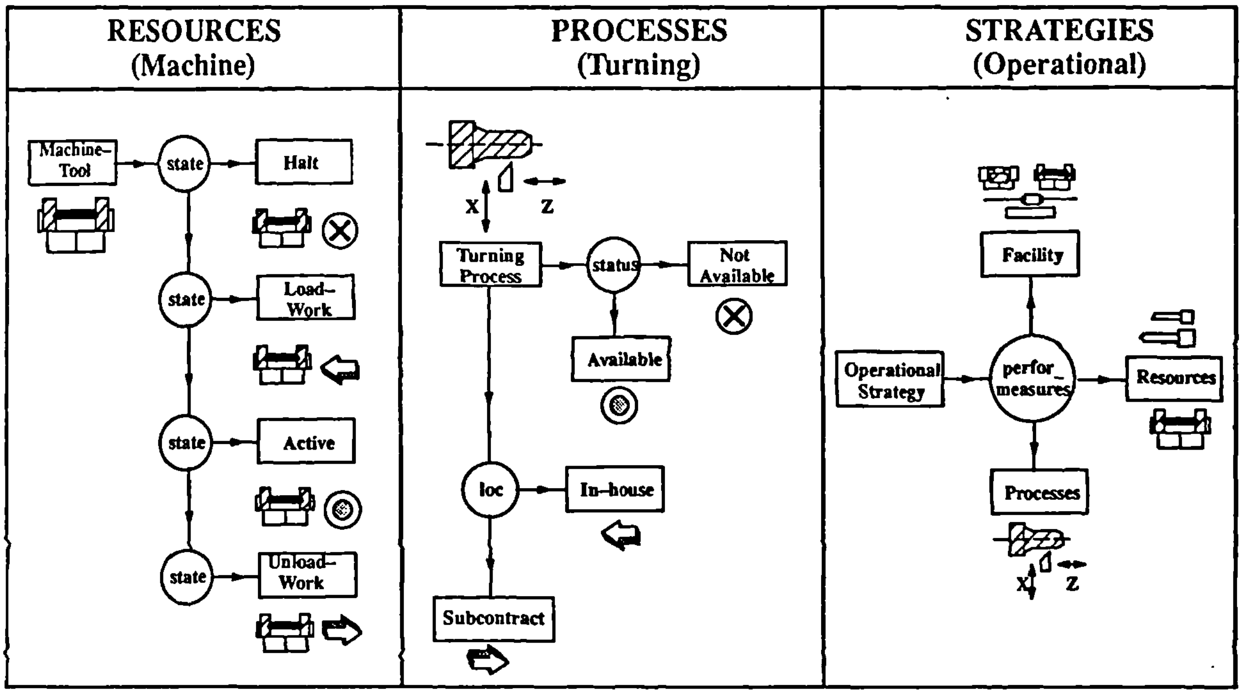
A. Molina	Figure 11.2 Manufacturing Information Modelling Methodology	LUT–SE Research Group



A. Molina | Figure 11.3 Modelling the Data Dimension of the Manufacturing Entities in the Manufacturing Model | LUT-SE Research Group



A. Molina | Figure 11.4 Modelling the Function Dimension of the Manufacturing Entities in the Manufacturing Model | LUT-SE Research Group



A. Molina	Figure 11.5 Modelling the Behaviour Dimension of the Manufacturing *Entities in the Manufacturing Model*	LUT-SE Research Group

GENERAL CAPABILITY INFORMATION

RESOURCES (Machine Tool)

Capabilities:

Product Size
Product Weight
Type of process
Type of materials
Tolerances
Surface Finish

PROCESSES (Turning Process)

Capabilities:

Type of materials
Type of form
Product Volume
Tolerances
Surface Finish

SPECIFIC CAPABILITY INFORMATION

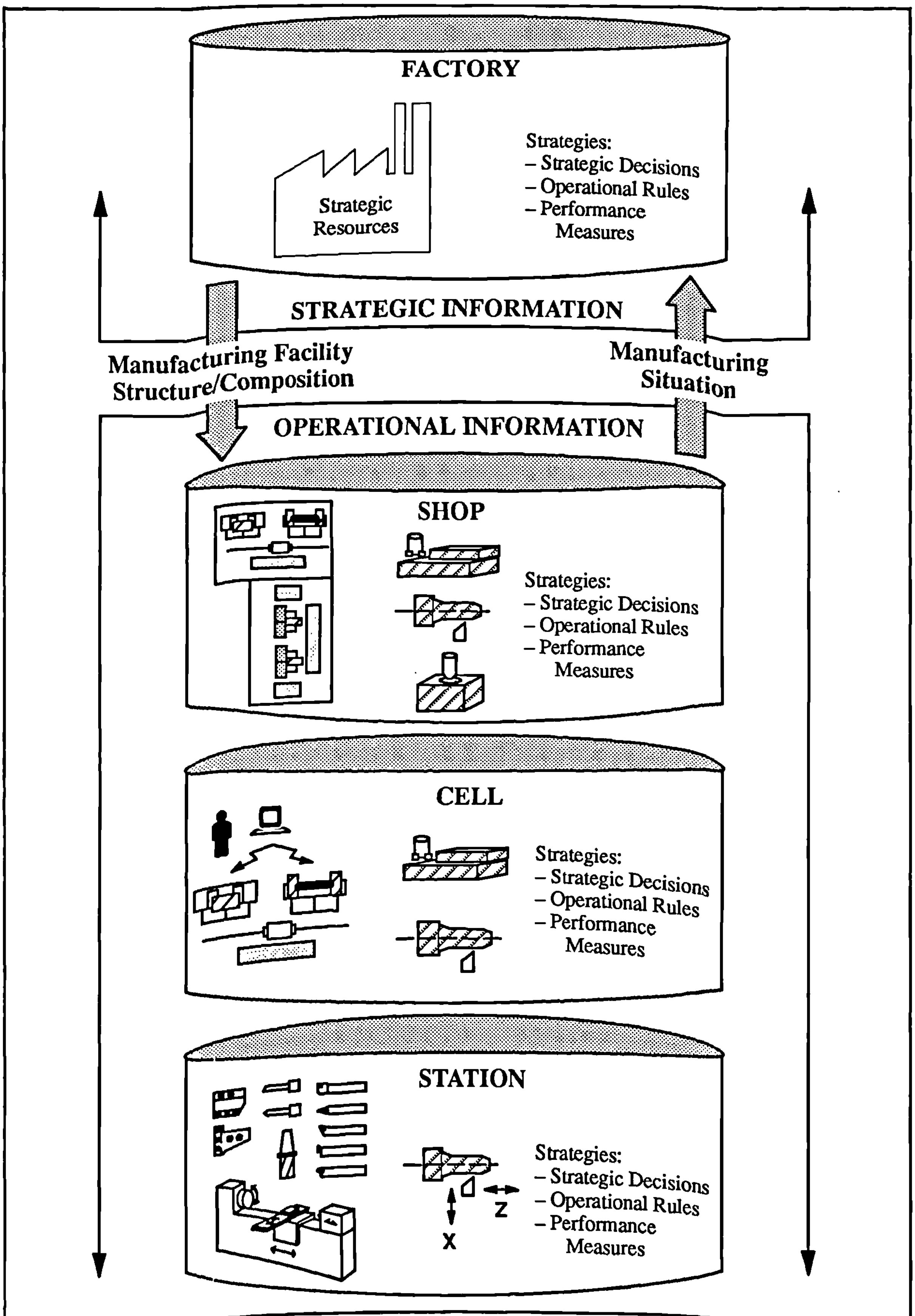
STRATEGIES (Operational)

Objectives:
Reduce Set-up Time

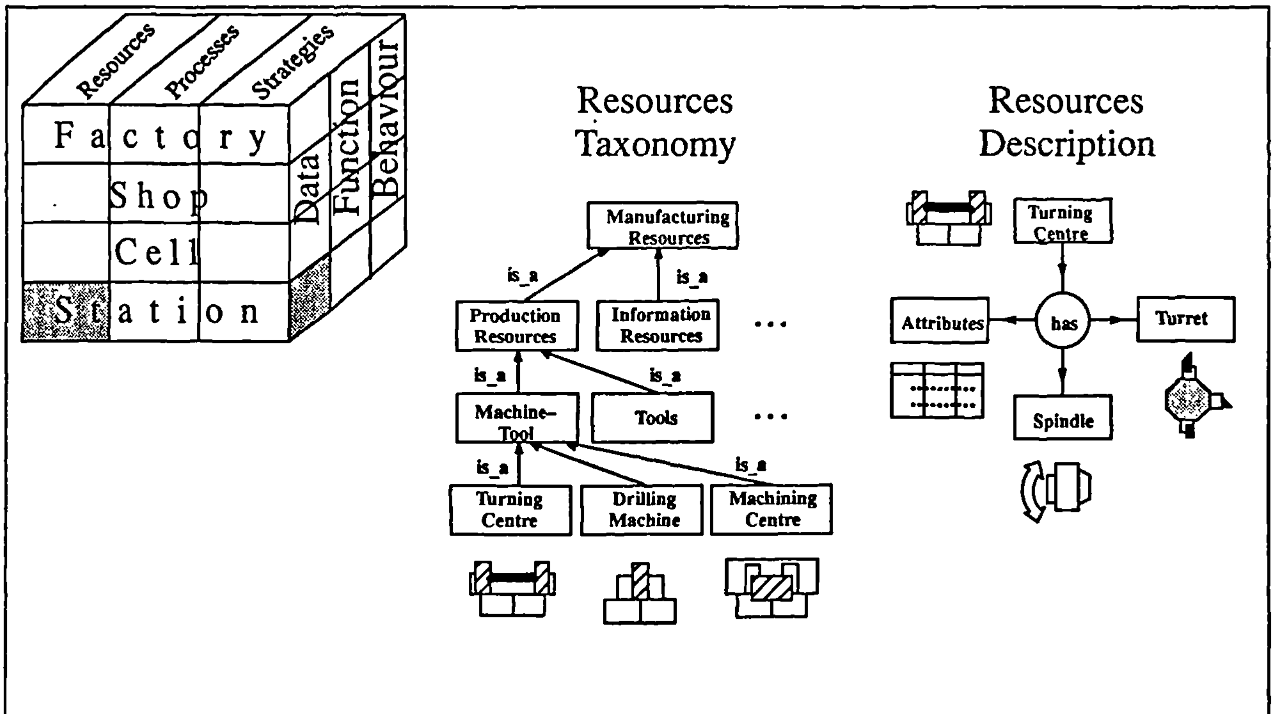
Decisions:
Use Machine Tool for dimensions in range [10 -100]

Rules:
IF dimensions > 100 AND
dimensions < 10
Machine Tool not adequate

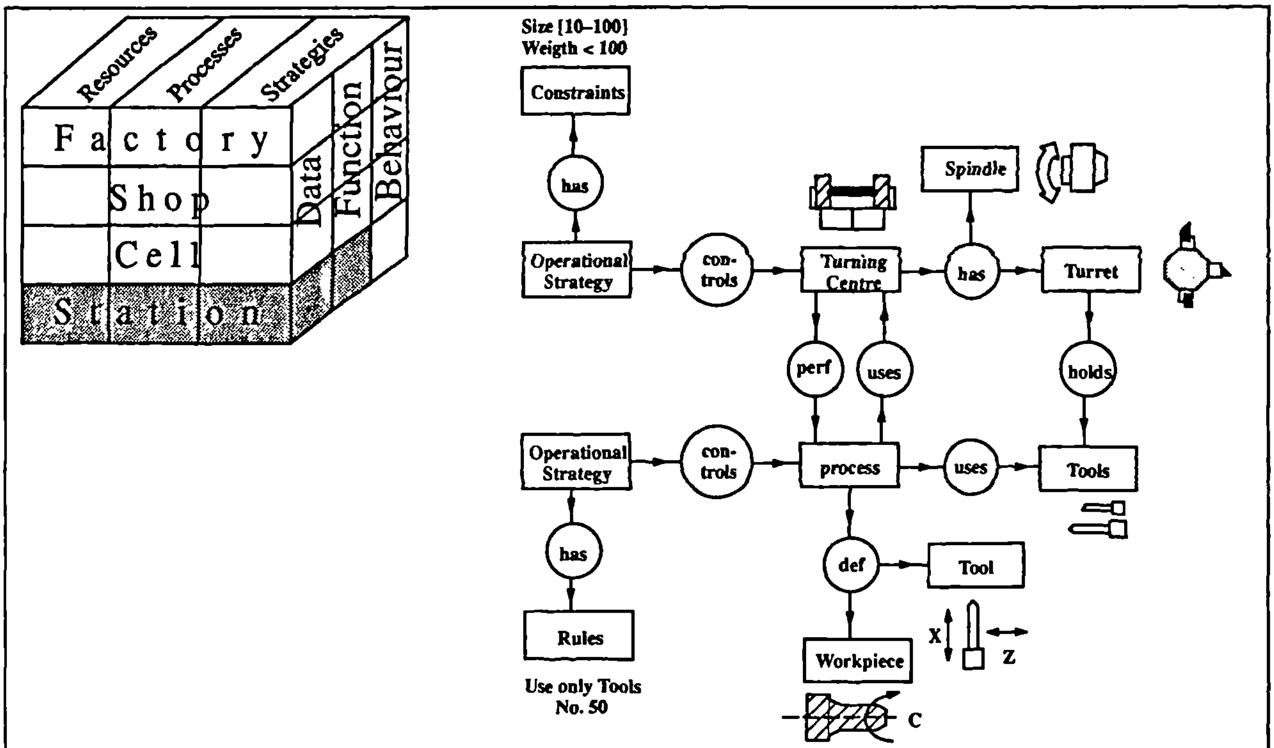
A. Molina	Figure 11.6 Describing General and Company's Specific Capability Information	LUT-SE Research Group



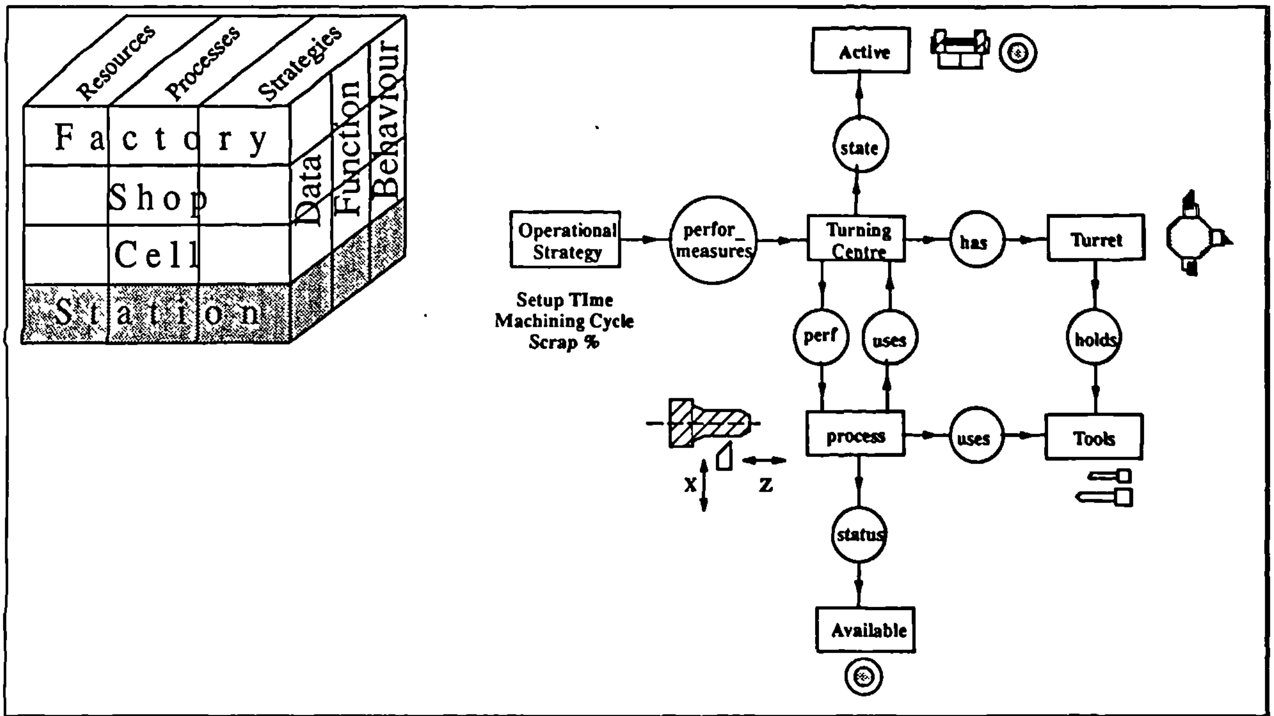
A. Molina	Figure 11.7 Manufacturing Model represents Strategic and Operational Information	LUT–SE Research Group



A. Molina	Figure 11.8 Data Modelling of Resources at Station Level	LUT-SE Research Group



A. Molina	Figure 11.9 Function Modelling of Resources, Processes and Strategies at Station Level	LUT-SE Research Group



A. Molina

Figure 11.10 Behaviour Modelling of Resources, Processes and Strategies at Station Level

LUT-SE Research Group

Chapter 12

The Exploitation of the Manufacturing Information Modelling Methodology

12.1. Introduction

This chapter introduces the modelling work carried out by the author in order to build up the Manufacturing Model. The use of the Booch Object Oriented Methodology as the underlying method is discussed. The important issues addressed by the author during the use of the modelling methodology described in Chapter 11 are presented. The generic classes created to represent the manufacturing resources, processes and strategies are described in detail. These generic classes are the foundation of the Manufacturing Model structure and content.

12.2. The use of Booch Object Oriented Methodology and EXPRESS Language

The author has chosen the Booch Object Oriented Design Method, as explained in Chapter 10 (Section 10.3), to be the underlying methodology for the development of the Manufacturing Model and guide the modelling of the data and function, in order to build an EXPRESS representation of the Manufacturing Model.

The Booch object oriented methodology conceives (Figure 12.1):

- two definition models: the logical and the physical, and
- two models of semantics: static and dynamic.

The author is mainly concerned with the logical and static models, where the representation of manufacturing entities it is possible by using Class and Object Diagrams. Class diagrams are used to represent the modelling dimension of data, function and behaviour. A class diagram represents entities that can be grouped into classes because of its similarities or common characteristics by using two relations, i.e. generalisation/specialisation (relation *is–a*) and semantic relations (*contains and uses*). The semantic relations in Booch are of two types *contains* (relation *has*) and *uses* (any semantic relation). Therefore the abstraction mechanism of generalisation/specialisation was defined using the Booch relation *is–a* (Figure 12.1). The abstraction mechanism of aggregation was implemented using the Booch semantic relation *contains* (*has*) and the other semantic relations needed to describe the function and behaviour dimension were defined using the Booch semantic relation *uses*. In addition, in Booch, a template is used to represent a description of the class, relations with its super–classes and subclasses, its relation with other classes and the relevant attributes that describes the class.

In order to be able to use the Booch methodology, the Manufacturing Information Modelling Methodology has been mapped into the Booch cube (Figure 12.2). This mapping enabled the author to use the Booch notation to model the different aspects required, i.e. data, function and behaviour. Figures 12.3, 12.4 and 12.5 illustrate how the Booch method has been used to model the different modelling dimensions described in the previous chapter in figures 11.8, 11.9 and 11.10. By using Booch as the methodology of the manufacturing information modelling work the author introduced a discipline in the development of the EXPRESS model.

The modelling of the manufacturing entities (i.e. resources, processes and strategies) to build the structure and content of the Manufacturing Model was carried out as follows:

1. Development of the taxonomies for manufacturing resources, processes and strategies based on the generalisation/specialisation relation *is–a*.

2. Description of class aggregation using the Booch semantic relation – *contains (has)*.

3. Definition of functional composition using the Booch semantic relation – *uses*

4. Specification of each class using an augmented Booch standard template, which includes an EXPRESS definition

Before starting to model each individual manufacturing entity, the concepts defined in the Manufacturing Information Modelling Methodology have to be translated into Booch notation definitions, in order to remain consistent with the methodology employed. This representation is illustrated in figure 12.6 where the Manufacturing Model *describes* a facility which is composed of resources, processes and strategies. Therefore, a Facility *has* resources, processes and strategies. In fact, all different types of facilities (e.g. stations, cells, shops and factories) will consist of these three manufacturing entities. A facility has been defined as any type of system which allows the manufacture of products, and can be classified according of how a manufacturing firm organizes their resources and processes. Thus in the Manufacturing Model, in accordance with the levels, the facilities can be stations, cells, shops or factories. This is described using the Booch notation in the following manner: factory *is–a* facility, shop *is–a* facility, cell *is–a* facility and station *is–a* facility (Figure 12.6). The use of the concept of inheritance (relation *is–a*) allows any subclass of facility to be composed of resources, processes and strategies, as well. Then a station *has* resources, processes and strategies, so the shop, cell and factory. In addition, because of the organization of facilities within a company, a factory *includes* shops, a shop *includes* cells and a cell *includes* stations. These relations enable the Manufacturing Model to represent complete manufacturing sites, by combining and reusing these definitions.

The author wants to make a clear distinction between the levels of the Manufacturing Model and the elements which are represented at those levels. A level describes a set of facilities and their capabilities, i.e. at the Station Level a set of stations can be described, and not necessarily just one station. This explicit difference allows the definition of different configurations of the Manufacturing Model by combining the necessary levels which are required to meet the information needs for a particular application environment. For example, the Design for Manufacture Environment, described in Chapter 5 Section 5.4, will only require the Station level of the Manufacturing Model.

The multi–level characteristic of the Manufacturing Model enables a suitable definition of strategic and operational information to be made. The manufacturing entities required to represent operational information for a particular facility can be represented at the Shop, Cell and Station Levels by describing shops, cells and station, respectively. For example, at the Station Level the description of a station with their associated resources, processes and strategies can specify completely the capability of that station. In a similar manner, at the Factory Level the representation of strategic information can be specified by associating a factory with the manufacturing strategies which are employed to decide how the factory has been organized and is currently being used.

12.3. The development of standardised taxonomies and generic classes

Once the Booch definition of the Manufacturing Model was made, the author concentrated on developing the classifications for resources, processes and strategies according to the guide-lines defined in the document of KCIM and MANDATE (ISO TC184 SC4 WG8/N9). This classification has been proposed for resources only, however the author has extended it to include processes and strategies. The classification allows to define Resource Groups, Generic Resources, Specific Resources and Individual Resources. In order to perform this task a template defining the manufacturing entities was created to describe the following information related to the resources, process and strategies: description, relationships, attributes and a formal EXPRESS definition (Figure 12.7). These templates have been very useful to characterise each manufacturing entity and standardise the documentation of the Manufacturing Model.

These templates were used by the author to created the classifications of resources, processes and strategies as follows:

- Resource Groups, Generic Resources, Specific Resources and Individual Resources.
- Process Groups, Generic Processes, Specific Processes and Individual Processes.

- Strategy Groups, Generic Strategies, Specific Strategies and Individual Strategies.

These classifications were constructed based on the relation *is–a* in order to organize the different type of manufacturing entities in hierarchies and define the suitable levels of abstraction to allow the taxonomy to be consistent and easily extensible. Each type of class is represented by a specific Booch notation in the following manner (Figure 12.8):

- Resources/Processes/Strategies Groups represented as metaclasses and classes
- Generic and Specific Resources/Processes/Strategies as parameterized classes
- Individual Resources/Processes/Strategies as instantiated classes.

A major issue in developing classifications (taxonomies) is to decide when a class need to be subdivided into further subclasses or not. Specially because a common problem when creating taxonomies is to over classify. In this research, the results of the first exercise in developing the resources taxonomy was a hierarchy with too many levels, which were not necessary. After a major revision the author, together with Mr. T.I.A. Ellis, developed a more suitable taxonomy according to the research objectives. This taxonomy helped the author to understand better the mechanism for creating classifications and lead to define an important characteristic of the Manufacturing Model, the capability to define Generic classes which could be used as reference to allow the users to create their own Specific and Individual classes.

Another important decision to be made when creating taxonomies is the one related to the use of multiple inheritance. Multiple inheritance allows a class to inherit characteristics from different classes, i.e. the ability to describe a class based on other classes. The author found very useful the use of multiple inheritance in cases where the classification of Groups can be made based on different criteria, and therefore, a Generic class is required to be defined from all these different perspectives. For example the classification of processes by material they work with (e.g. solid, liquid, gas), type of process they perform (mass conserving, mass reducing, etc.) and by energy they required (mechanical, chemical, thermal).

Once the taxonomies where created the process to define the Generic classes for resources, processes and strategies was undertaken. The issues to address here are related to the creation of Generic classes based on the aggregation relation *has*, and the description of their function and behaviour using semantic relations such as: *uses*, *performs*, etc.

The approach taken by the author was to perform a bottom up analysis of the taxonomies in order to build up the Generic classes using the aggregation relation *has*. For each Generic class the author defined the set of components that the particular class may required. For example to define a *Turning_Centre* class one should describe the turning centre based on the component which are common to all turning centres such as: turret, spindle, bed, etc. The results of this aggregation is the definition *Turning_Centre has Turret, Turning_Centre has Spindle,* etc. It is possible to group all the Turning Centre components under a class called *Turning_Centred_Component*, therefore the generalization of this definition is *Turning_Centre has Turning_Centre_Component*, in the same manner a relation can be created to define *Milling_Centre has Milling_Centre_Component*. Even more, this generalization process can be taken further up in the hierarchy by defining the following relation *Machine_Tool has Machine_Tool_Component*. Where all the possible types of machine tool components are grouped under the class *Machine_Tool_Component*. The same criteria was applied to define all the aggregations, wherever possible, required to describe the complete set of generic classes of resources, process and strategies.

From the experience learned by the author when building aggregation relations *has*, the author decided to use a top–down approach, instead of a bottom–up, to describe the functional capabilities and behaviour of the Generic classes. The top–down approach seemed to be more efficient than the bottom–up approach, especially because less time was spent on deciding the types of relations needed to describe the function and behaviour. The decision of defining semantic relations at the higher level possible, i.e. at the class located closer to the top of the taxonomy, was important in order to accelerated the development of definitions. The top–down approach facilitates the definition of function and behaviour because it was possible to define semantic relations which could be applied to a wide range of classes. The author decided this in order to simplify this task and to focus in the classes that were required to be defined in greater detail. Example of this relations are: *Manufacturing_Resource performs Manufacturing_Process* and *Manufacturing_Process uses Manufacturing_Resource.* It is important to mention that by defining these semantic relations at high level, there is a need to verify and denote which relations are not possible at lower levels. For example, the inheritance allows the definition of a turning tool performing production processes, which is not true. On the other hand, a turning process, which is a manufacturing process, and thus it uses production resources, in this case the turning tool which is true. In these cases a special validation process should be performed. A bottom–up approach was used to carry out this validation, where remarks where added to the definition indicating when a specific semantic relation was not applicable. The advantages gaining by defining the semantic relations which represent function and behaviour at the higher levels allowed the author to reuse these relations to describe complex representation, which will be explained later in this chapter for each type of manufacturing entity.

Finally, the identification of the important information attributes associated to each Generic class was performed following a simple but effective criteria: define only the attributes which were required in the CIMOSA model and therefore the ones to be used by the data–driven applications. This decision was important in order to simplify the quantity of information to be represented, because one may decide to include attributes which probably will not be needed. Well defined CIMOSA models and IDEF0 models are key to identify which information will be required. However attributes can be added whenever they are needed, however the generic structure of the Manufacturing Model will remain the same. In order to represent these attributes in a standard form a template was defined which includes: the superclasses and subclasses of the class being defined, its relations with other classes, and its EXPRESS definition. In the next subsections issues addressed during the modelling of resources, processes and strategies are described.

12.4. Modelling Manufacturing Resources

The manufacturing resources are all the physical elements within a facility that enable the product realization such as: production machinery, production tools, material handling equipment, storage systems, humans, supply and disposal units, etc. The resources can be organized in groups to create manufacturing facilities such as stations, cells or shop floors.

Early in this research, the author recognised the need to represent the resources in a function oriented manner in order to describe their role in supporting the design and manufacturing activities. Thus, the description of the resources are based on their physical properties and functional composition which allows to capture their capabilities.

The author's work on modelling resources has been influenced by different elements of other related research:

- KCIM (ISO TC184 SC4 WG8/N9): Similar classification of Resource Groups.
- CIMOSA (Esprit 688/5288): Classification of Generic Resources in two major classes Passive and Active Resources. Generic Resources are also need to be categorised according to four criteria: the possibility of being moved, the possibility of being scheduled, the possibility of being replicated, and the possibility of being shared.
- IMPPACT (Gielingh and Suhm 1993): Specification of the attributes of Generic Resources in terms administrative, economical and technical attributes.
- ISO and BSI Standards (Molina 1993): Use of standard classification codes for Generic Resources, specially tool descriptions.

The Resource Groups defined in this research were: Furniture and Fittings, Human Resources, Information Processing Resources, Material Handling Resources, Measuring and Testing Resources, Production Resources, Supply and Disposal Units and Storage Resources. However taxonomies were developed only for the following groups:

1. Production Resources
2. Supply and Disposal Units
3. Material Handling Resources
4. Storage Resources

Figure 12.9 shows a partial example of the Resource Groups and taxonomies. To be able to create the taxonomies different books, handbooks, journal papers and catalogues were consulted by the author and T.I.A. Ellis. The complete taxonomies are documented in the MOSES document (moses–core–mm–3), and have not been included in this PhD thesis do to its length. However some examples are presented throughout this chapter to illustrate the relevant aspects of the taxonomies.

12.4.1. Production Resources

In this research the production resources have been defined to be the resources that are required for processing supplies, work in progress and products. Production resources are automatically controlled (by other manufacturing resources), or/and operated by a human. All production resources consist of supplies, i.e. components. The machine is formed of components (e.g. bed, tool carrier, etc.), so the tool assemblies which may have tool components, like screws and retention knobs, besides production tools. The following two main classes of production resources were defined:

1. Production Machinery is non movable machinery that is for processing work in progress. There are two subtypes: Discrete Part Machinery and Continuous Part Machinery. Figure 12.10 represents part of the production machinery taxonomy. At the lower level the class machine tool is defined, other examples of production machinery are assembly lines, welding equipment, ovens, etc. Figure 12.11

illustrates how the different types of Booch classes are used to define the generic, specific and individual resources.

2. Production Tools are movable equipment that must be present while processing work–in–progress. There are 5 subtypes of production tools: processing tool, tool assembly, tool guide, tool holder and workholding tool. Examples of these tool are: fixtures, tool guides, cutters, hand tools, a tool assembly containing an adapter and a processing tool, etc.

In order to build the taxonomies of production machinery and tools SME Handbooks (Dallas 1976, Drozda and Wick 1989), manufacturing books (Moore and Kibbey 1982, Kalpakjian 1984, Doyle 1985, Schey 1987, El Wakil 1989) and manufactures' products catalogues (e.g. Traub, Mazak, Sandvick, Komet, Stellram) were consulted. Once an agreement was reached by the author and T.I.A. Ellis on the taxonomies for production resources, the definition of the aggregation relation *has* was carried out. The aggregation relations *has* defined in the production resources are:

1. Machine_Tool *has* Machine_Tool_Components

2. Tool_Assembly *has* Tool_Holder, Processing Tool, and Tool_Component

3. Modular_Holder *has* Basic_Holder, Intermediate_Adaptor and Tool_Adaptor

4. Indexable_Insert_Cutter *has* Insert_Holder and Insert

The relation *has* between the Machine_Tool and its Machine_Tool_Components allows the definition of a variety of Machine Tool by combining different component (e.g. bed, spindle, turret, tailstock, etc.). The generic machine tool class is described in figure 12.12. An advantage of the object oriented approach is that the inheritance mechanism can be used to enable complex entities to be defined by combining several simpler entities. It is therefore possible to define the capability of a range of multi–axis turning centres by associating a number of machine tool components (e.g. spindle, turret, tailstock, etc.), see figure 12.13

The definition of a Tool_Assembly can be made by combining the classes associated to the Tool_Assembly class. Figure 12.14 illustrates a Tool_Assembly composed by tool holders (e.g tool adaptors), *processing tools (e.g. drills), and required tool component* (i.e. screws, sleeves). In a similar manner the class Modular_Holder allows the description of complex modular tooling systems by grouping the elements which are required to build up a holder. For example in figure 12.15 the modular tooling system represented has basic holders (e.g. DIN 2080. Yamazaki, VDI2814 and ANSI), intermediate adaptors (e.g. extensions), and tool adaptors. The last definition *has* represents the nowadays popular indexable insert cutters, these cutters are formed by combining an insert holder with the suitable insert. The figure 12.16 shows some examples of these type of cutters.

The aggregation relation is not enough to represent the functionality of the production resources. In order to be able to describe the function there is the need to use the following semantic relations: *holds, performs* and *controlled_by.* The author would like to point out that semantic relations are not only between manufacturing resources but between resources and processes, and resources and strategies. The semantic relations used to represent the functionality of production resources are:

1. Tool_Carrier *holds* Production_Tool, Cutter_Holder *holds* Cutter, Cutter_Driver *holds* Generic_Tool_Holder, Insert_Holder *holds* Insert, Arbor *holds* Adaptor, Stud *holds* Adaptor, Extension_Bar *holds* Adaptor, Basic Holder *holds* Intermediate_Adaptor and Tool Adaptor.

2. Manufacturing_Resources *performs* Manufacturing_Processes

3. Manufacturing Resources *controlled_by* Manufacturing_Strategies

The *holds* relation allows the description of the capability of tooling system, how different cutters and holders can be combined to be used in a manufacturing process, or to represent tool assemblies. In this research the author has concentrated in machining process, and the tooling systems used in performing turning, drilling and milling operations in turning centres. Figures 12.17 through 12.21 highlight some of the possible tooling systems which can be described using this relationship.

The semantic relation *performs* enables the definition of the capabilities of manufacturing resources in terms of what manufacturing process they can carried out. For example modern turning centre can perform turning, milling processes, and drilling as shown in figure 12.22. It should be noted that in this example the turning centre is represented by its functional components, i.e. spindle, turret and tailstock. On the other hand, as mentioned in section 12.3, production tools can not perform processes, therefore the relation *performs* does not apply to them. Here the definition taken from CIMOSA between active and passive resources is used to make a distinction between production machinery which are active resources, and production tools which belong to the type of passive resources. It is important to highlight this difference because the relation *performs* for production tool has to be specified as not applicable. The same relation *performs* is used to describe the possible processes carried out by Material Handling Resources (see subsection 12.4.3)

The description of how a resource is used and organised is made by using the relation *controlled_by* between the manufacturing resources and manufacturing strategies. The rules imposed by company's decisions on how to utilize a particular resource are described through the usage of this relation. More will be said about manufacturing strategies later in Section 12.6. At this point the reader should only be aware that a link can be made between resources and the rules which control their usage, as shown in figure 12.23.

Finally the description of the attributes for each generic class have to be made. This is not an easy task because a lot of information can be incorporated into the model. First of all the attributes were classified in three groups: administrative, technical, and operational attributes. The following information is represented in these groups:

- Administrative data is related to the suppliers of the resources, the resource information required to acquire.

- Technical: data concerned with the type, geometry, composition and properties of the production resources.
- Operational: data associated with individual production resources regarding temporate conditions of the production resources such as: planning data, usage and real time management.

These type of information has been translated from an informal description to an EXPRESS description as shown in Table 12.1 and Table 12.2.

12.4.2. Supply and Disposal Units

The supply and disposal units are the components which constitute major resources, such as: machine tools, production tools. Two classes have been defined: machine tool components and tool components. The importance of these generic classes rely on their ability to be grouped to form a resource. In this manner a complex resource can be defined based on the sum of its components. This characteristic has been exploited in this research to increase the flexibility of the Manufacturing Model by allowing a diversity of machine tools and production tool to be configured.

12.4.3. Material Handling Resources

There are a wide variety of equipment types available. Apple (1972) published that the number of available options in the major categories was a total of 570 types of material handling equipment. Advances in technology in recent years, especially in the area of automation, have resulted in an even greater number of options. Equipment types that are being increasingly applied today that were not considered by Apple include automated guided vehicles (AGV), industrial robots and car–on–track conveyors. However, only a subset of these equipment types is commonly used within a given application area such as in–plant handling of discrete parts, bulk material handling, or handling in storage and warehousing. In this research in order to be able to develop a suitable taxonomy the following sources were consulted: Apple (1972), Tompkins and White (1984), Kulweic

(1984), and various journal papers (Material Handling Engineering Reference Guide 1988, Matson et al. 1992, Dowlatshahi 1994). Based on these contributions a limited taxonomy was developed. This taxonomy only considers the AGVs, Positioners (Robots), Conveyors, Monorails, Cranes and Industrial Trucks (Figure 12.24).

The most important relation considered for Material Handling Resources is *performs*. This relations enables to represent the capability of these type of resources to perform transportation processes. By using this semantic relation the representation of an AGV moving parts or tools can be made. In the same manner the capability of a robot to load and unload a machine tool can be easily represented (Figure 12.25).

The attributes of the Material Handling Resources are:

- Administrative data is related to the suppliers of the resources.
- Technical: capabilities to transport different sizes and weights, and the equipment speed.
- Operational: status

12.4.4. Storage Resources

The storage resources includes the following elements: buffers and automated storage and retrieval systems (Ranky 1983, Hartley 1984). The buffers can be automatic pallet changers and temporary storage. The automated storage and retrieval systems includes a wide range of automated warehouses, pallet stockers, etc. These resources have a very simplified representation with the following attributes:

- Administrative data is related to the suppliers and information required to maintenance.
- Technical: capabilities in terms of number of pallets it can store, maximum weight and size of the pallets.
- Operational: status

All of the resources described above have an EXPRESS definition. The complete EXPRESS model is documented in the MOSES document (moses–core–mm–4), and have not been included in this PhD thesis do to its length.

12.5. Modelling Manufacturing Processes

The term process can in general be defined as a change in the properties of an object, including geometry, hardness, state, information content (form data), and so on (Alting 1982). To produce any change in property, three essential agents must be available: material, energy and information. Depending on the main purpose of the process, it is either a material process, and energy process, or an information process. In manufacturing these three type of process exists, nevertheless this research only considers one of them: material processes. The material processes used in a facility and performed by resources need to be represented based on their capabilities. The representation of the capabilities of material processes, such as turning, drilling, milling and assembly processes is of major importance.

The author's work on modelling processes has been influenced by:

- Alting (1982) classification of Process Groups by material they work on, energy they use and type of process.
- BSI DD 203:Part 1 (1991) classification of Process Groups by function they perform transformation, transportation, storage and inspection.

The taxonomy of processes is based on the above classifications. Figure 12.26 illustrates a section of the taxonomy, where multiple inheritance is used to define the properties of the processes. For example, the Turning process *is_a* Solid_Material_Process because it works on solid materials, *is_a* Mechanical_Process because its energy flow is mechanical, and finally *is_a* Chip_Forming_Process because produces chips and therefore *is_a* mass reducing process (i.e. transformation process).

Different generic representations were analysed and evaluated in order to determine the best way to describe a manufacturing process at a higher level of abstraction. In particular, the Enterprise Activity (CIM–OSA) and the Generic Activity Model (BSI DD 203: Part1 : 1991). A generic definition of the process based on the Generic Activity Model, which can easily be mapped into the Enterprise Activity construct of CIMOSA, has been defined in the Manufacturing Model. Using this description a representation of most manufacturing processes in terms of inputs/outputs/controls can be made.

In addition to the representation of manufacturing processes in a high level structure, the author has focused on representing the capabilities of the chip forming processes (machining processes). These capabilities are constrained by the capabilities of the resources which can perform the machining process. Therefore, the author decided to represent these capabilities by using relations between the possible movements between the machine components (spindle, turret, work table, etc.) and the production tools (tool holders and cutters). A similar approach is used by Gindy et al. (1994) to represent process capability models for equipment selection.

Chip forming processes are characterised by the relative motions of a cutter against a workpiece. A specification for a process is built when defining an instance of a process. For example, a Turning process can be defined that will inherit all the properties of a Chip Forming Processes and hence allow the definition of general properties e.g. tolerance, surface finish, etc. There may be several specifications for the process but each will define tool types and the relative motion required between the tool and the workpiece. In the example represented by figure 12.27, the turning process requires the free rotation of a workpiece, and static single point tools which can move along the axis X and Z. The same figure illustrates the drilling process with two possible representation: the workpiece locked and a live drill, or the workpiece in free rotation and a static drill. The drill in both cases will require movements in X and Z.

The resource section of the Manufacturing Model can therefore be searched for resources that are able to meet the process specification. In this case, the resource selected is a turning centre (Figure 12.28). The turning centre has both a spindle which can rotate and the capability to hold a workholding device (i.e. chuck). It also has an indexable turret which can move in the X and Z whilst holding the suitable cutters in a cutter holder. This definition of process can be generalized to include other types of chip forming process such as: drilling, milling, etc.

The last important relation of the process is the one with strategies (*controlled_by*). The strategies will define how the process can be used, for example which materials can be employed (Figure 12.29). It is important to mention that even when the turning process is capable of machining other type of materials for this particular instance only cast iron, steel and casting should be used, i.e. this is a rule of thumb of the company.

The same classification for the attributes of resources is used for process, i.e. administrative, technical, and operational attributes. Table 12.3 and Table 12.4 show how these attributes have been translated from an informal description to an EXPRESS description.

12.6. Modelling Manufacturing Strategies

The representation of structured resources and processes allows the Manufacturing Model to have a reliable representation of the manufacturing facilities and their capabilities in terms of process technology and equipment. In addition to this type of information, there is a need to represent the manufacturing strategies, because the strategies are decisions made on the use and the organization of resources and processes (e.g. constraints imposed on the use of a certain type of resource or process). There are two types of decisions which make possible the formulation of manufacturing strategies: decisions made over time which define the structure, capacity and technology of the facilities, and the day to day decisions which determine how to use the facilities and related processes. In the Manufacturing Model, strategies will represent how the resources and processes are structured and used to support the realization of the manufacturing function in order to achieve the manufacturing objectives of a company.

Manufacturing strategies are functional strategies where the principal focus is on the maximization of resource productivity (Hofer and Schendel 1978). Hayes and

Wheelwright (1984) defines that " .... a manufacturing strategy consists of a sequence of decisions, that over time, enables a business unit to achieve a desired manufacturing structure, infrastructure, and set of specific capabilities." Therefore manufacturing strategies can be defined as the pattern of structural an infrastructural decisions which determines the strategic capabilities of a manufacturing organization.

Relevant to this research on modelling strategies, from the eight classes suggested by Hayes and Wheelwrigh (1984), the author has considered only strategies related to facility structure (focus), capacity, technology and production planning/material control (Figure 12.30).

The research on Strategic Manufacturing Decision Support (SMDS), undertaken by Dr. W. Wei, has been a major influence to the work presented in this section. The SMDS objectives are (1) research the linkage between business strategy and manufacturing performance; (2) research a strategic manufacturing decision–aid for top management and (3) investigate the generation of factory performance requirements for middle management. Discussions were carried out between the author and Dr. W. Wei to define the information structure of a strategy. The objective was to find a representation of strategies which could be common to both projects. It was agreed to have the following representation of a strategy in the Manufacturing Model (Figure 12.31):

1. Strategic Decisions: variables which indicates a choice in a particular matter. They are related to four categories: focus, capacity, technology and Production Planning/Material Control.

2. Operational Rules: rules with the aim to support the strategic decision. These rules are based on the employees experience and company know–how.

3. Performance Measures: important parameters which allow the evaluation of a decision and related operation rule(s). These measures belong to one of the following categories cost, quality, flexibility and delivery.

Each level of the Manufacturing Model represents strategies associated to particular instances of a type of facility. The characteristics of these three elements of the information structure are defined in figure 12.32. For each strategy category (i.e. facility focus, capacity, technology and production planning/material control) a set of pre–defined variables has been defined to describe a strategic decision. For example in the facility focus the following variables have been specified: Product Group, Product Volume, Product Life Cycle and Production Mode. The values that each variable can take determine a decision regarding an important aspect of the facility. In the next sections these variables are explained in detail. In relation to the operational rules, the ones at the Factory Level are usually generic, i.e. non associated to any specific process or resource. For example, the policy (operational rule) of a factory could be reduce cost whenever possible. This operational rule is used as the guide for other operational rules defined in the lower levels of the Manufacturing Model. For example, a rule that could support this high level rule could be " use turning centre X to reduce cost in tooling for turning operations". Therefore at the lower levels the rules are always associated to a particular resource or processes. The performance measures are not predefined due to the fact that each company has its own performance metrics. Performance measures in general as they extend down through the organisation must become increasingly specific, they must encompass shorter planning horizons and some must emphasize on cost performance (Keegan et al. 1989). The following subsections describe in detail the categories of strategies developed in this research.

12.6.1. Facility Strategies

Facilities strategies are related to define how to specialize or focus each facility (Skinner 1974). Facilities may be focus by geography (location), product group, process type, volume, or stage in the product life cycle. Developing a well–thought–out strategy for facility focus automatically provides guidance to the firm in determining the size, location, and capabilities of each facility. The following strategic decisions variables and values were chosen in this research:

- Product group: (OKP) One of a Kind Products, (MP) Multiple Products, (FMP) Few Major Products, and (CP) Commodity Products (Hayes and Wheelwright 1984).
- Product Volume: (LV–LS) Low Volume – Low Standardization, (LV) Low– to medium Volume, (HV) High– to medium Volume, and (HV–HS) High Volume – High Standardisation (Hayes and Wheelwright 1984).
- Product life cycle: (NP) New Products, (GP) Growing Products, (MP) Mature Products and (DP) Declining Products (Horne 1987).
- Production Mode: (P) Project–job shop, (F) Functional–batch, (C) Cellular, (L) Line, and (C) Continuous (Lee 1992).

The values associated to each variable correspond to the decisions which can be made regarding a particular facility. For example, if a machining station (a type of facility) has a very specialized machine only dedicated to produce a specific product, then the Product Group variable will be one of a kind products, i.e. it is only focused on one type of product. Following this example, the Product Volume will be Low volume (LV), if this product is only produced now and then. Depending on where the product is situated in the market place the Product Life Cycle variable could be a Mature Product, if the product is well established. Finally Production Mode is the fundamental arrangement and method for manufacture, therefore the Production Mode, of this example, will be Project.

12.6.2. Capacity Strategies

The capacity strategies define the production capacity of a facility and how this capacity is managed (e.g added, reduced, allocated). Production capacity can be changed in a variety of ways, and it is sometimes difficult even to define or measure capacity. Capacity is determined by the plant, equipment, and human capital managed by the company. Important decisions include (Fine and Hax 1985, Kerr and Greenhalgh 1991):

- Product Range: (H) Highly Standard, (S) Standard, (LS) Low Standard and (C) Customized.

- Cycle Demand (related to how to deal with cyclical demand): (HE) Holding Excess Capacity, (HI) Holding Seasonal Inventories, (O) Overtime, and (S) Subcontracting.

- Increase Capacity (how to add capacity in anticipation of future demand): (O) Overtime, (IS) Increase Shifts, (S) Subcontracting, (EF) Enlarge Facility and (AF) Additional Facility in a new location.

Similar to the Facilities Strategies these variables can take values depending on each company decision. If the same example of the previous subsections is taken, then the Product Range will be Customized, Cycle Demand could be Overtime and Increase Capacity will be one of Overtime or Increase Shifts.

12.6.3. Technology Strategies

The technology category includes decisions regarding the technology that is incorporated in manufacturing processes and specific pieces of manufacturing equipment, the degree of automation in the production and material–handling processes (Configuration), and the connections between the different production stages (Layout). The following categories have been defined:

- Processes: list of processes that are included in a facility and that will be controlled by the operational rules.

- Resources: list of resources that are included in a facility and that will be controlled by the operational rules.

- Configuration: type of facility (HA) Highly Automated, (PA) Partial Automated, (LA) Low Automated and (M) Manual. Together with decisions made on Processing System (e.g. transfer line – automated, operator–assisted NC machines, multi–machine FMS, integrated CNC machines, etc.), Material Handling System (e.g. operator–assisted MHS, computer–controlled MHS, integrated MHS with AS/RS, etc.), Tool Handling System (e.g. manual changeover of tools, automated tool delivery to machines, computer–controlled tool migration, etc.), Quality Assurance System (e.g. manual off–line inspection, automated on–line inspection, feedback for automatic process control, etc.), Storage and Warehouse System (e.g. AS/RS, operator–assisted S/WS, etc.) and System Integration (e.g. integration of multiple NC, integration with CAD/CAM, integration with CAD/CAM and MRPII, etc.). These definitions are taken from Naik and Chakravarty (1992).

- Layout: type of intracell and intercell layout (Arvindh and Irani 1994). The intracell layouts can be (L) Line Layout, (U) U Layout, (S) S Layout, (LL) L Layout, (O) O Layout, (D) D (robot served) Layout and (W) Layout. The intercell layout are (L) Linear Layout, (X) X Layout, (T) T Layout, (N) Network of Cells, (CF) Common Facilities, (CC) Cascading Cells, (PF) Parallel Flow–lines, and (VC) Virtual Cells.

For each facility there must be a list of processes and resources defining its capability. Each resource and process will have a set of operation rules which will control their use. For example if a machine tool is only going to be used for turning processes then this is a rule which controls the use of that particular machine tool. The other two decision variables decide important aspects of the facility related to its functionality and supporting manufacturing function. The case studies presented in Chapter 13 (Subsection 13.2.5) and 15 (Section 15.5) will clarify the use of these strategic decisions, as it is easier to demonstrate their use with a real example.

12.6.4. Production Planning/Material Control Strategies

The choice of Manufacturing Planning and Control System (MPC) can be designed to reflect the particular needs of a business. Therefore the different decisions can be captured to represent the choices made in this area (Berry and Hill 1992, Maull et al. 1990):

- Master Production Scheduling: (ATO) Assemble to Order, (DTO) Design to Order, (ETO) Engineered to Order, (MTO) Make to Order, and (MTS) Make to Stock.

- Shop–floor Control: Push type and Pull type.

- Material Planning: Time–phased and Rate–based

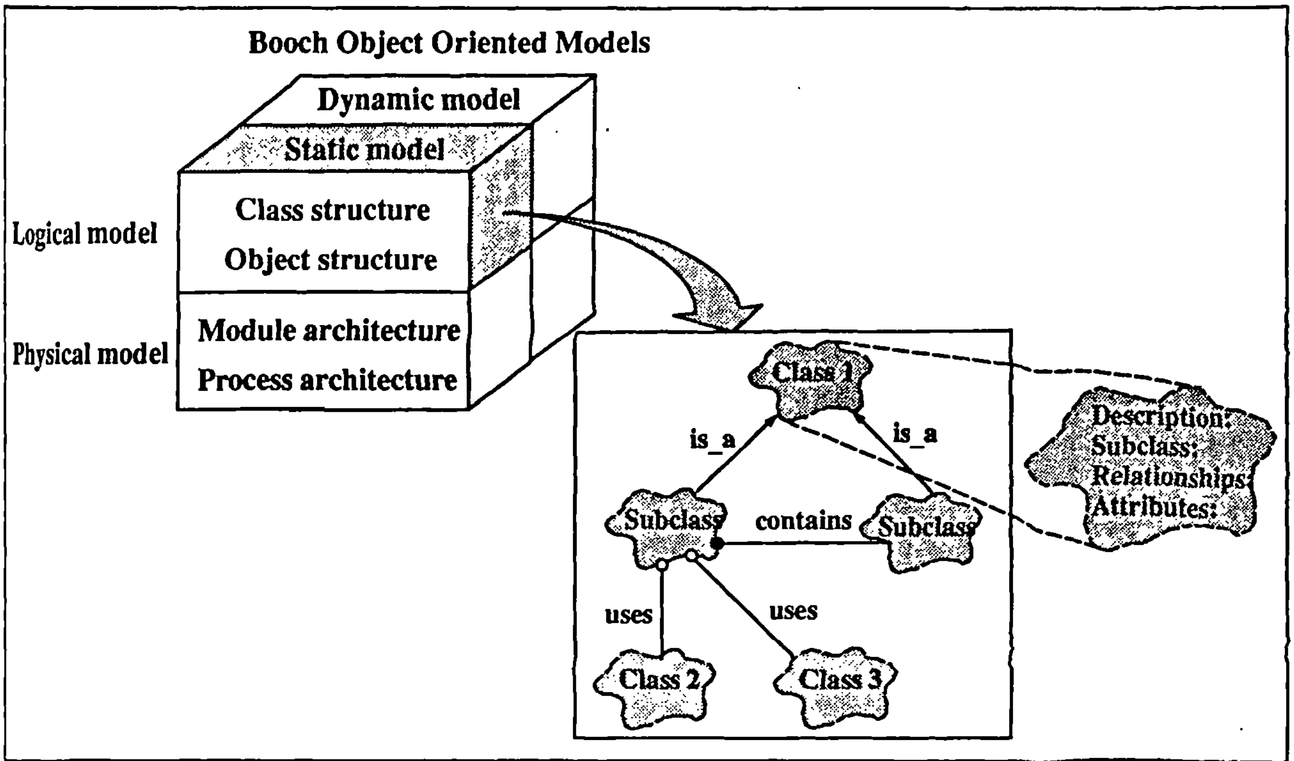
- Facility Scheduling: no predefined values.

These decisions are influenced by market characteristics and manufacturing facility focus, and are associated to each type of facility (e.g. station, cell or shop floor). Examples of how these variables are instantiated can be found in 15 (Section 15.5) .

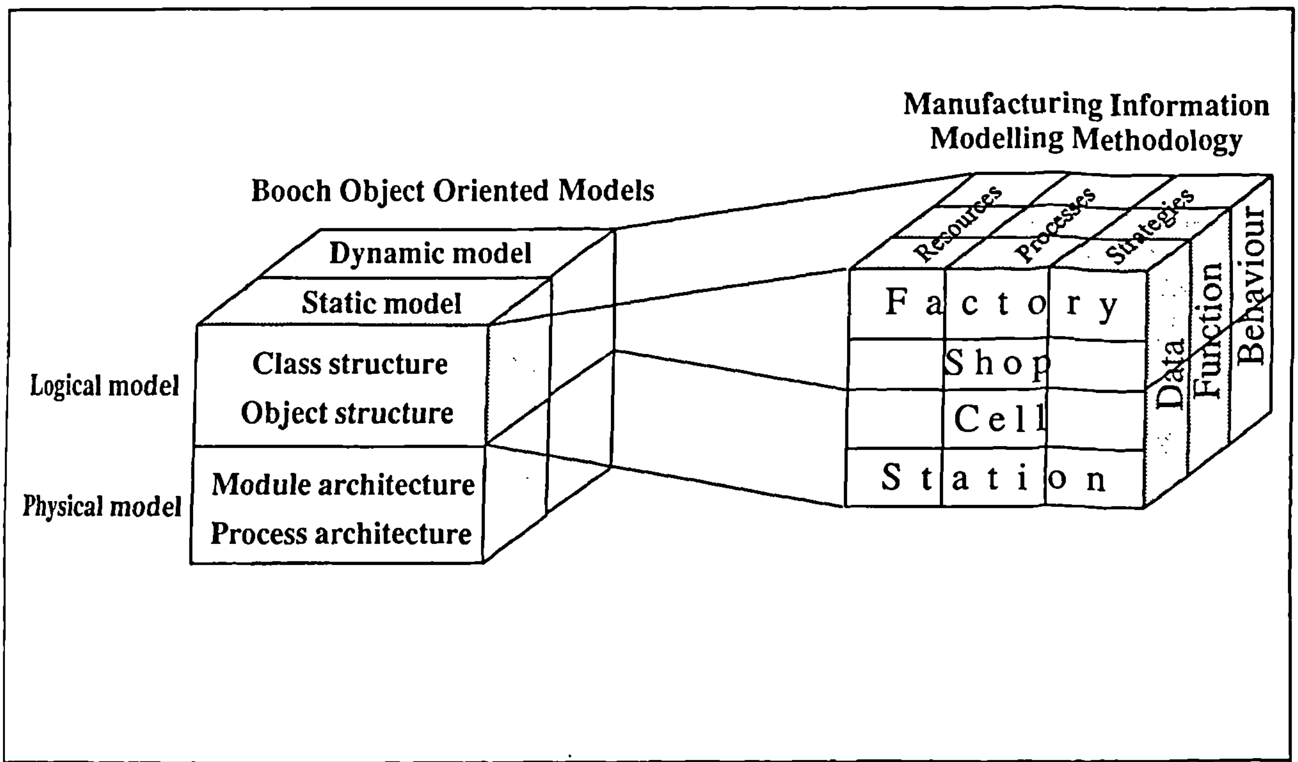
The complete representation of all the strategies described in this subsection is in Table 12.5 and the related partial EXPRESS model is presented in Table 12.6.

12.7. Modelling Manufacturing Facilities: Stations, Cells, Shops and Factories

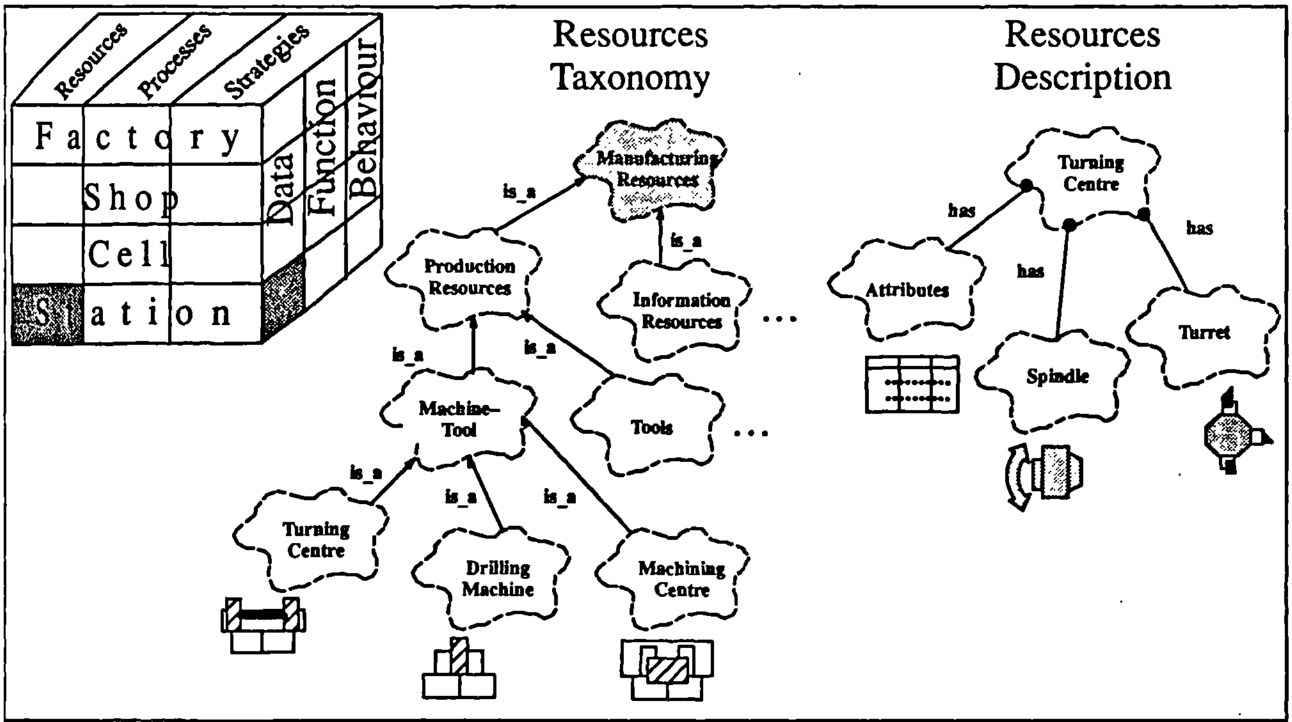
The modelling of stations, cells, shops and factories is carried out by using all the classes defined in the previous sections. The definition of different types of facilities is possible by re–using those generic classes. A station, cells or shop floors are considered types of facilities which have associated to it resources, processes and strategies. Therefore a diversity of these facilities can be created by instantiating the appropriate level of the Manufacturing Model. In this research no pre–defined facilities definitions have been made. Thus any type of station, cell or shop floor can be modelled. It should be noted that at the Factory Level of the Manufacturing Model there will be only strategies defined, as this level is intended to support the formulation of the Manufacturing Strategy of a company. The chapters 13, 14 and 15 illustrate the use of the Manufacturing Model by using two case studies: one related to the representation of a Rotational Parts Station (Station Level) and the other to a Rotational Parts Line and Assembly Line (Cell and Shop Level).



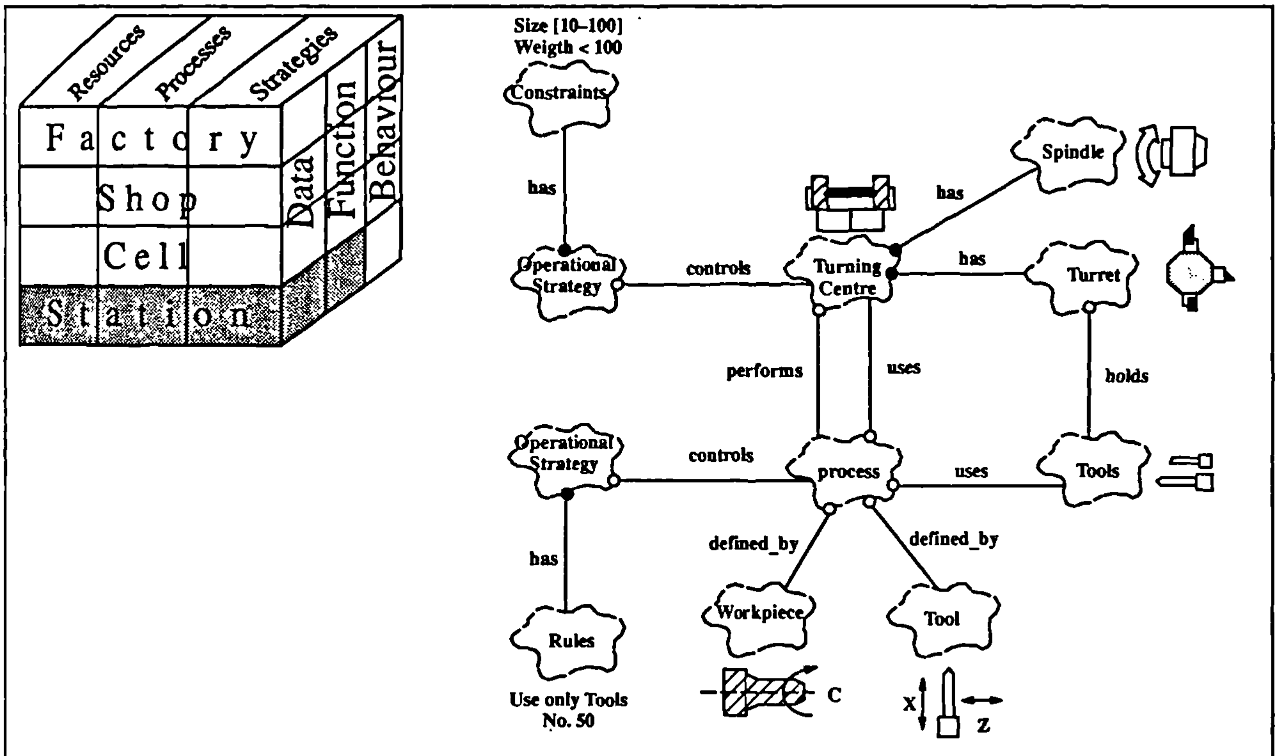
A. Molina	Figure 12.1 The Models of the Booch Methodology and Class Diagrams Notation	LUT-SE Research Group



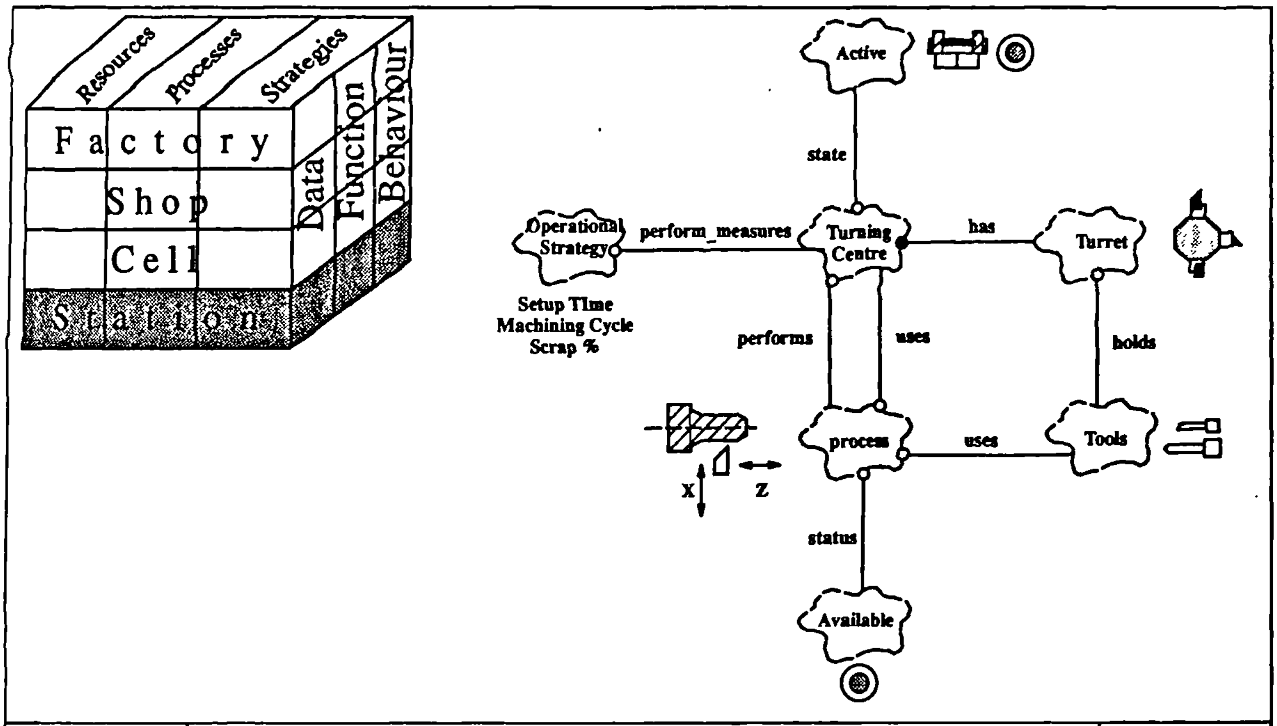
A. Molina	Figure 12.2 The use of Booch's Static and Logical Models in the Manufacturing Information Modelling Methodology	LUT-SE Research Group



A. Molina	Figure 12.3 Data Modelling of Resources at Station Level	LUT-SE Research Group



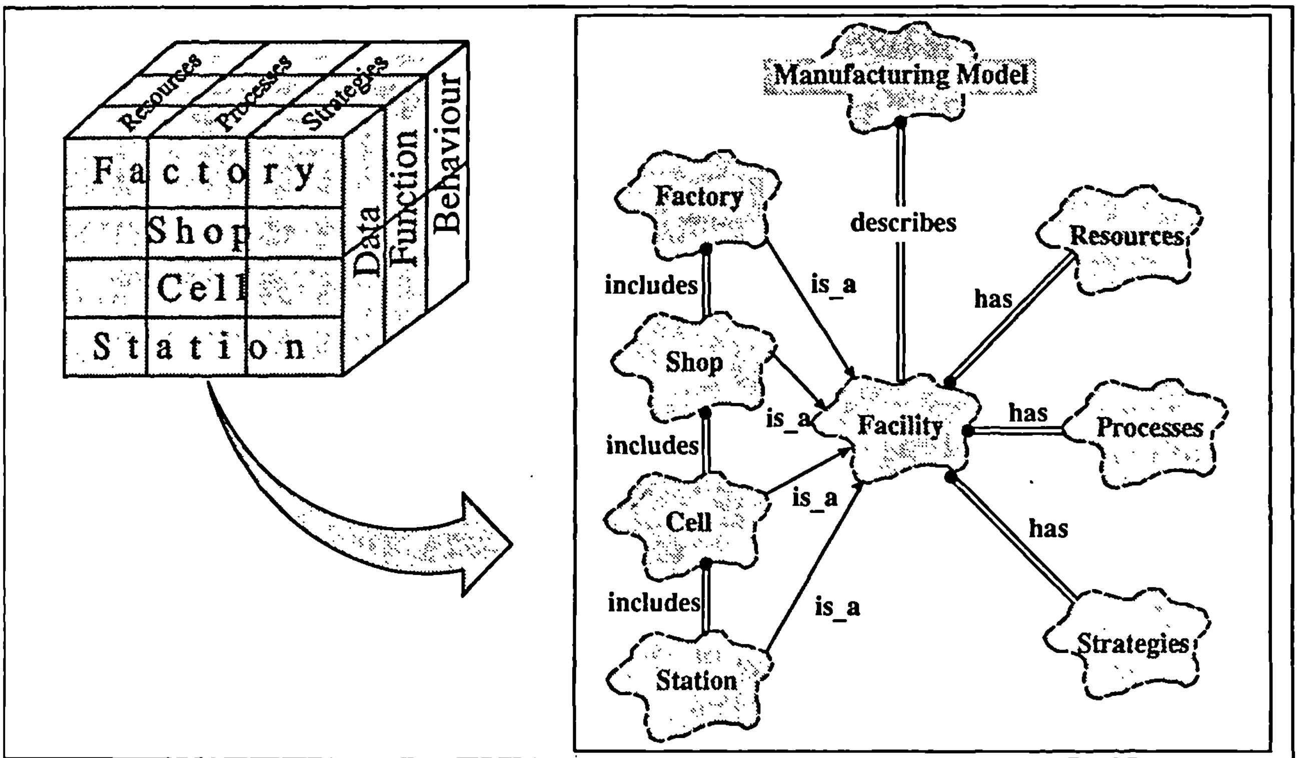
A. Molina	Figure 12.4 Function Modelling of Resources, Processes and Strategies at Station Level	LUT-SE Research Group



A. Molina

Figure 12.5 Behaviour Modelling of Resources, Processes and Strategies at Station Level

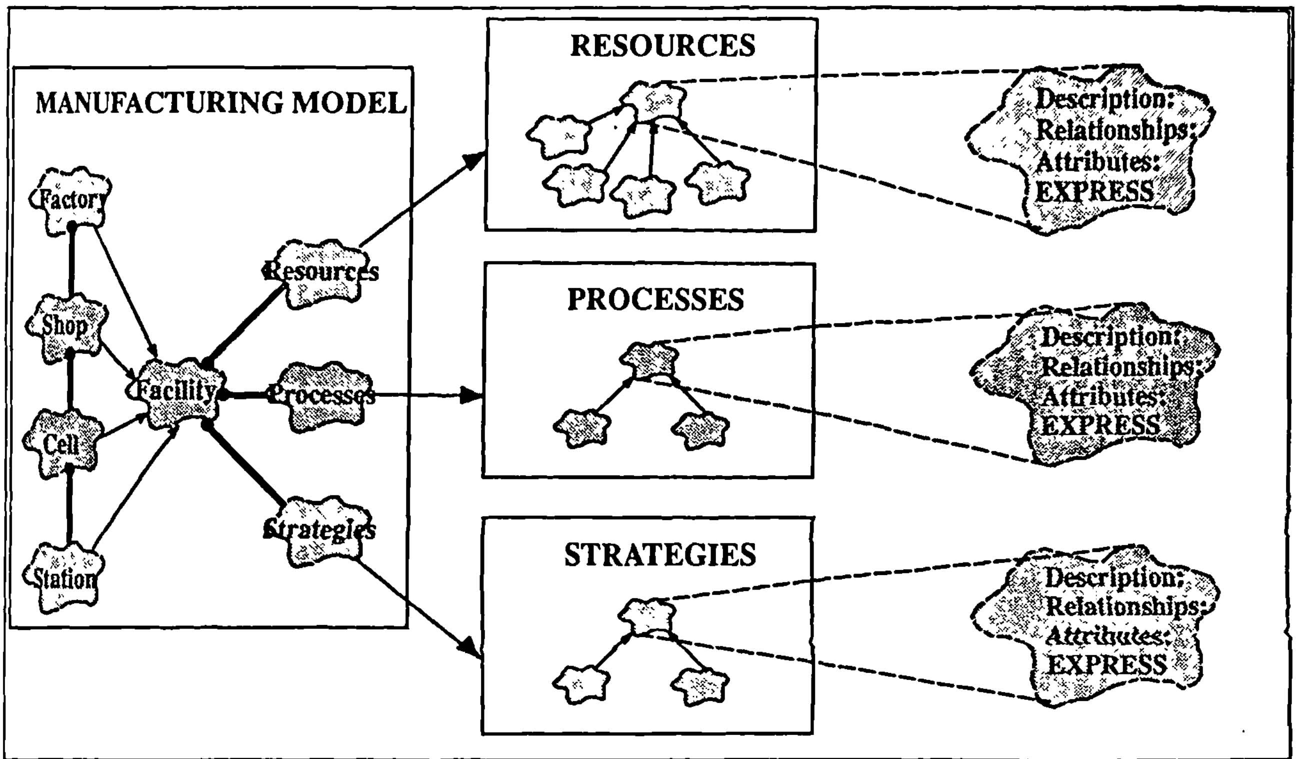
LUT–SE Research Group



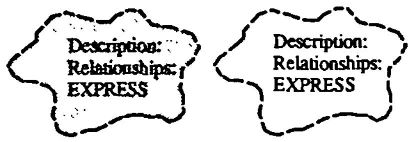
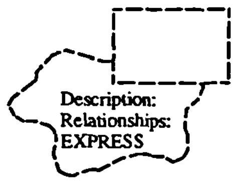
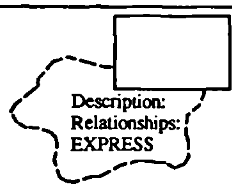
A. Molina

Figure 12.6 Representation of the Levels and Entities in the Manufacturing Information Modelling Methodology using Booch Notation

LUT–SE Research Group

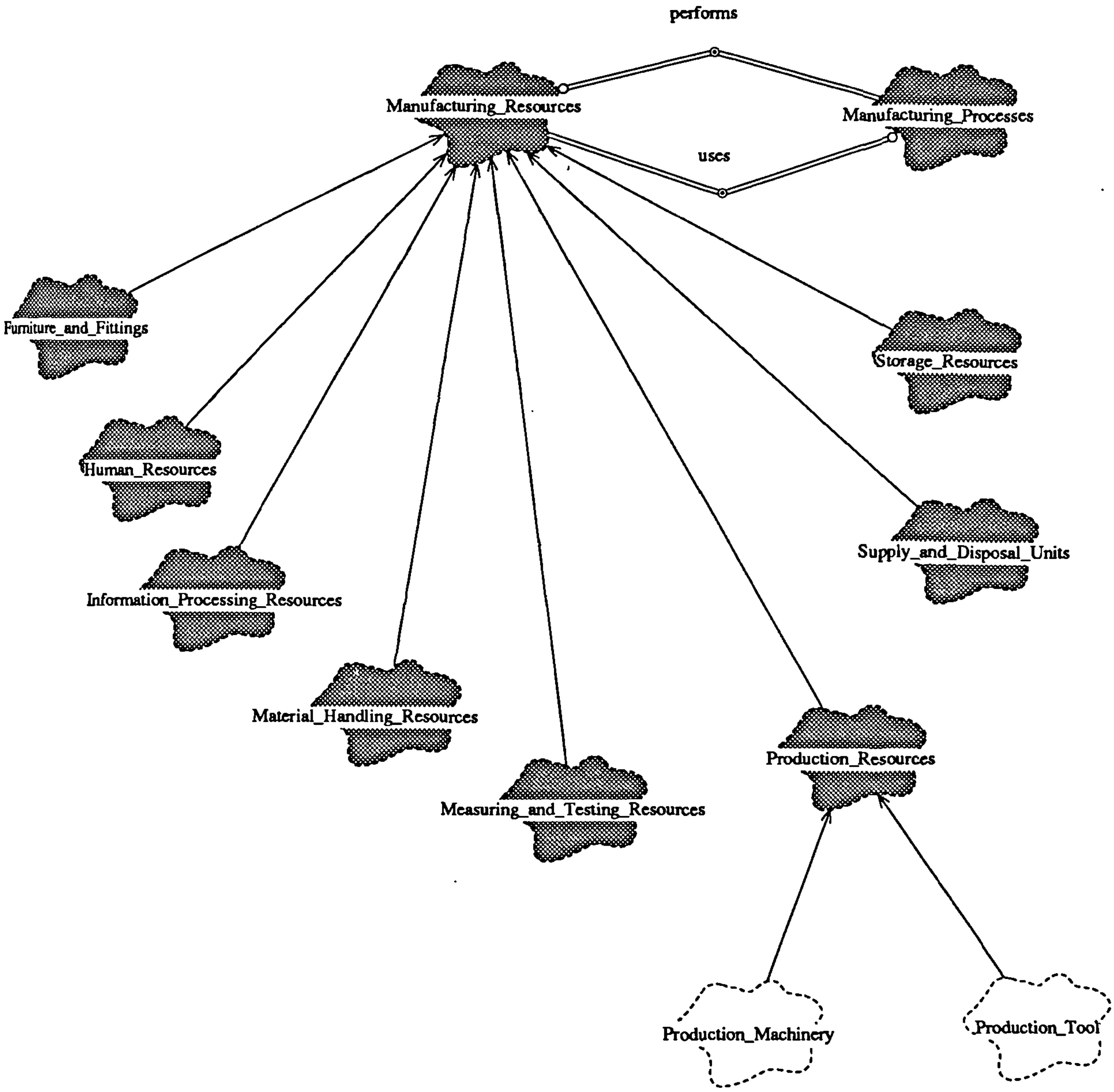


A. Molina	Figure 12.7 Taxonomies of Manufacturing Resources, Processes and Strategies	LUT-SE Research Group

Classification Based on ISO/TC184/SC4/WG8			Booch Class Notation
Resource	Process	Strategy	
Group: Production Resources Production Machinery Discrete Machinery Machine Tool	Group: Manufacturing Process Mass Reduccion Process Chip Forming Proces	Group: Manufacturing Strategies	 Metaclass Class
Generic: Turning Centre	Generic: Turning Process	Generic: Technology Strategies	 Parameterized Class
Specific: ST_40N_ATC_MC TRAUB HEIDS500	Specific: OD Turning Face Turning	Specific: Strategic Variable: OD Turning (ODT) Rule: Use only automated Turning Centres for ODT	
Individual: ST_40N_ATC_MC at Worcester Factory HEIDS500 at NEI	Individual: OD Turning in ST_40N_ATC_MC OD Turning in HEIDS500	Individual: Use ST_40N_ATC_MC to reduce set-up times Use HEIDS500 to reduce set-up times	 Instantiated Class

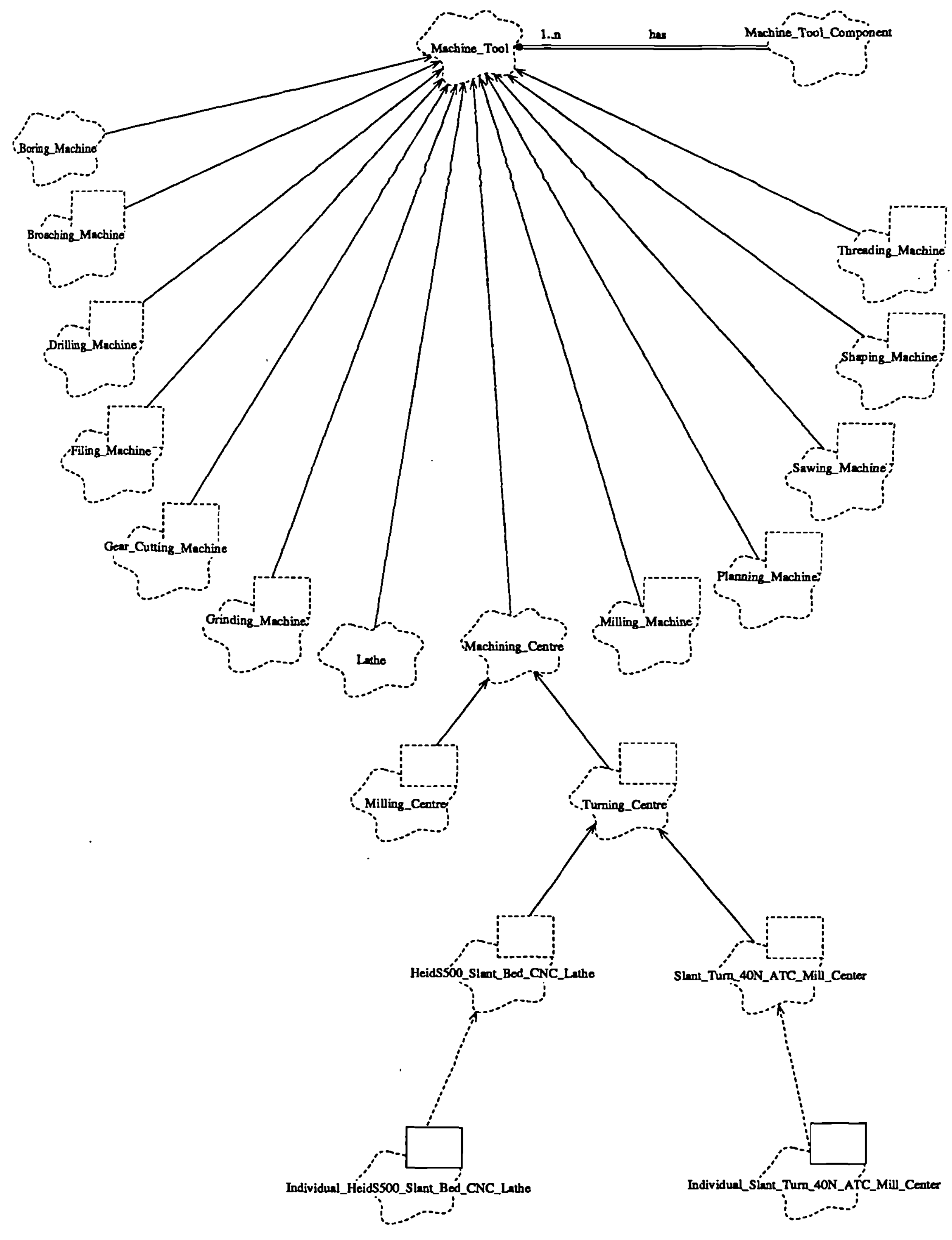
A. Molina	Figure 12.8 Types of resources, process and strategies represented by different Booch classes	LUT-SE Research Group

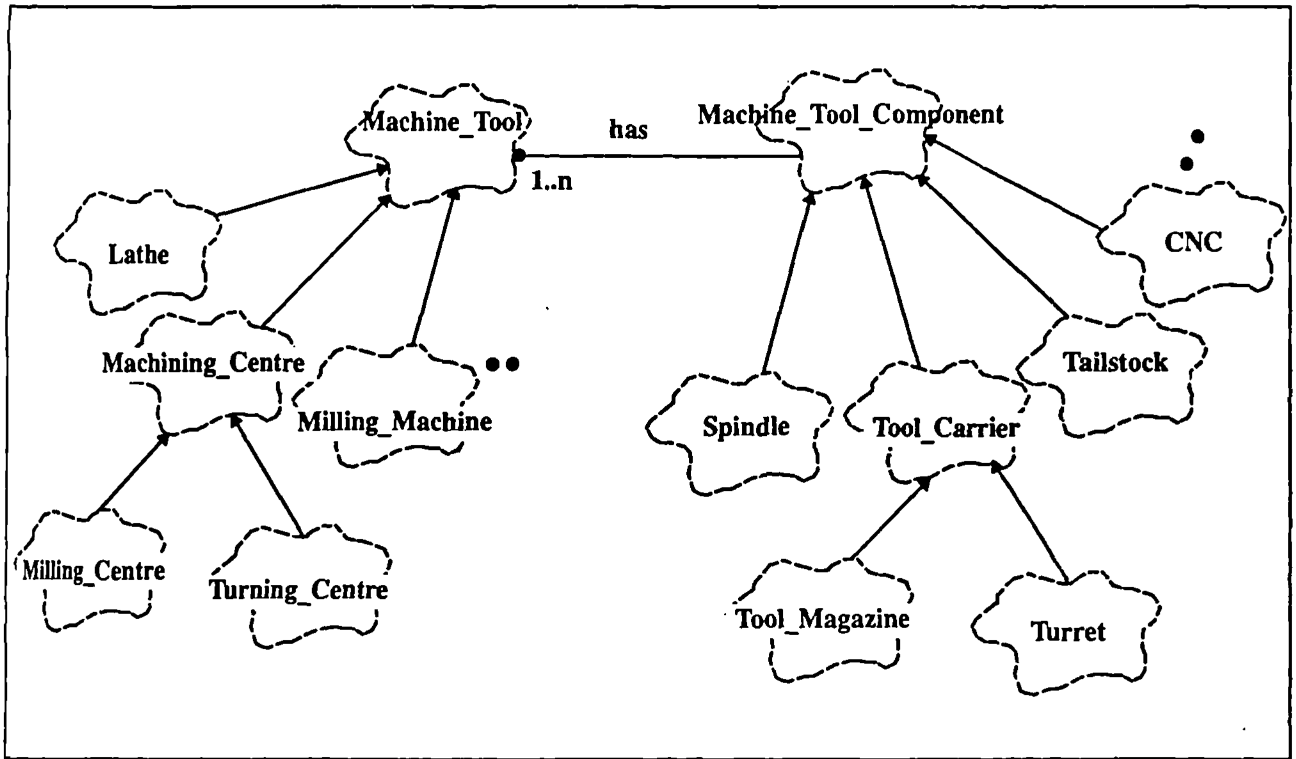
Project Team:	LUT-SE Research Group
Title:	Figure 12.9 Manufacturing Resources Taxonomy
Project:	Manufacturing Model
Printed by:	A. Molina



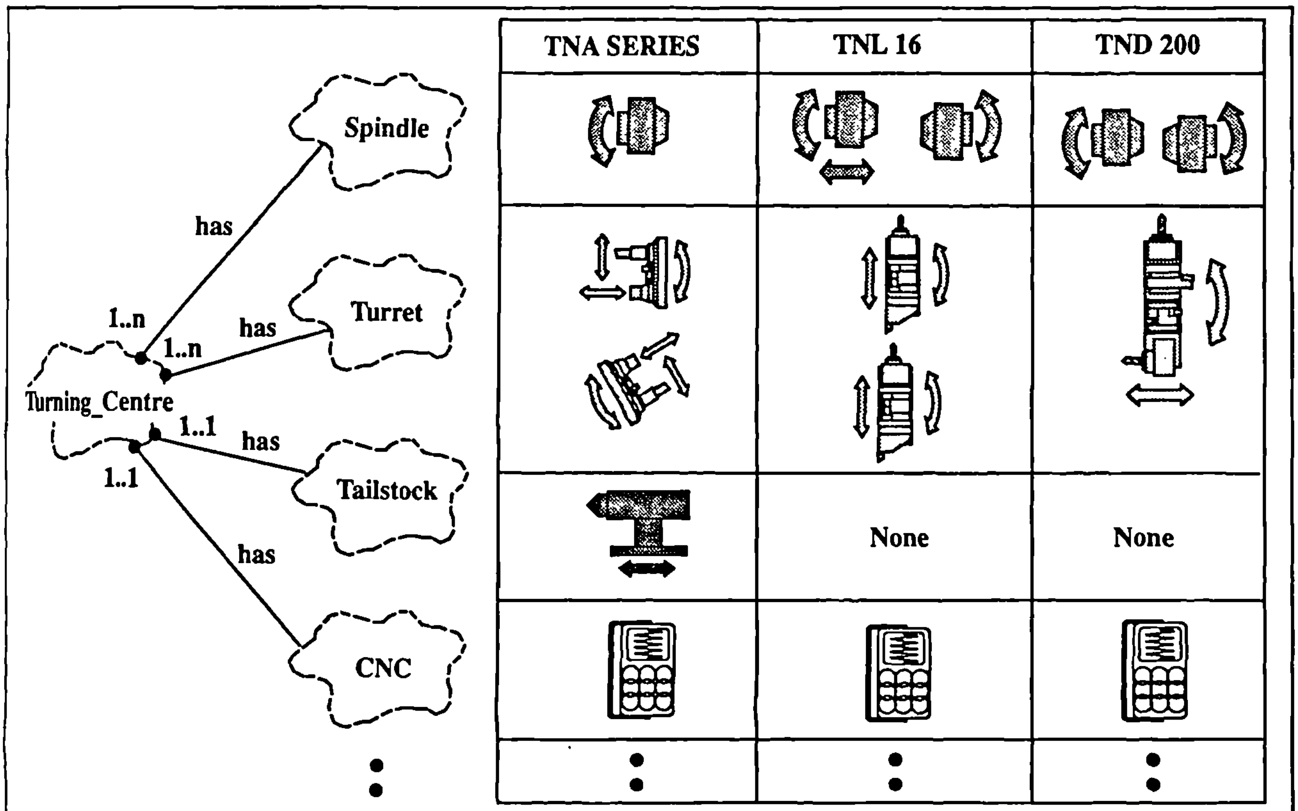
Project Team:	LUT-SE Research Group
Title:	Figure 12.10 Production Machinery Taxonomy
Project:	Manufacturing Model
Printed by:	A. Molina

Project Team:	LUT-SE Research Group
Title:	Figure 12.11 Machine Tool Taxonomy
Project:	Manufacturing Model
Printed by:	A. Molina

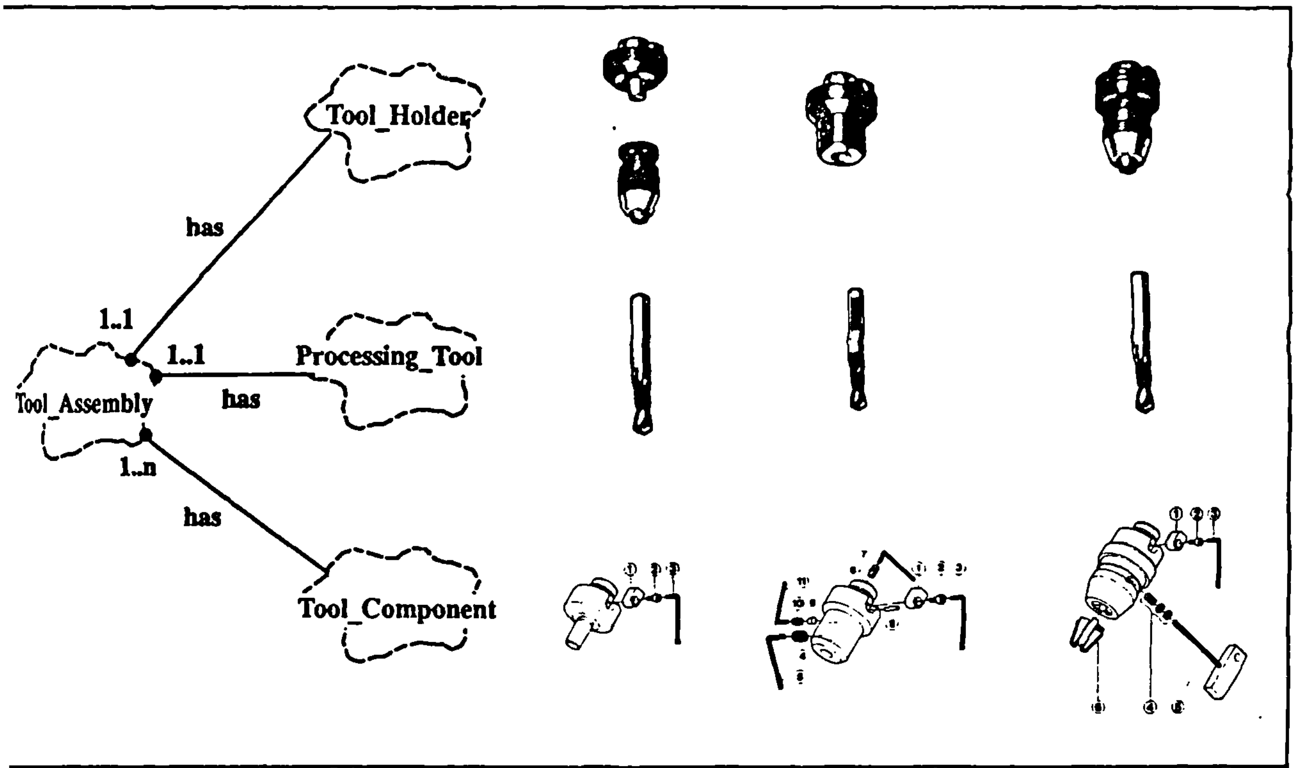




A. Molina | Figure 12.12 Modelling a Machine Tool and its Components | LUT-SE Research Group



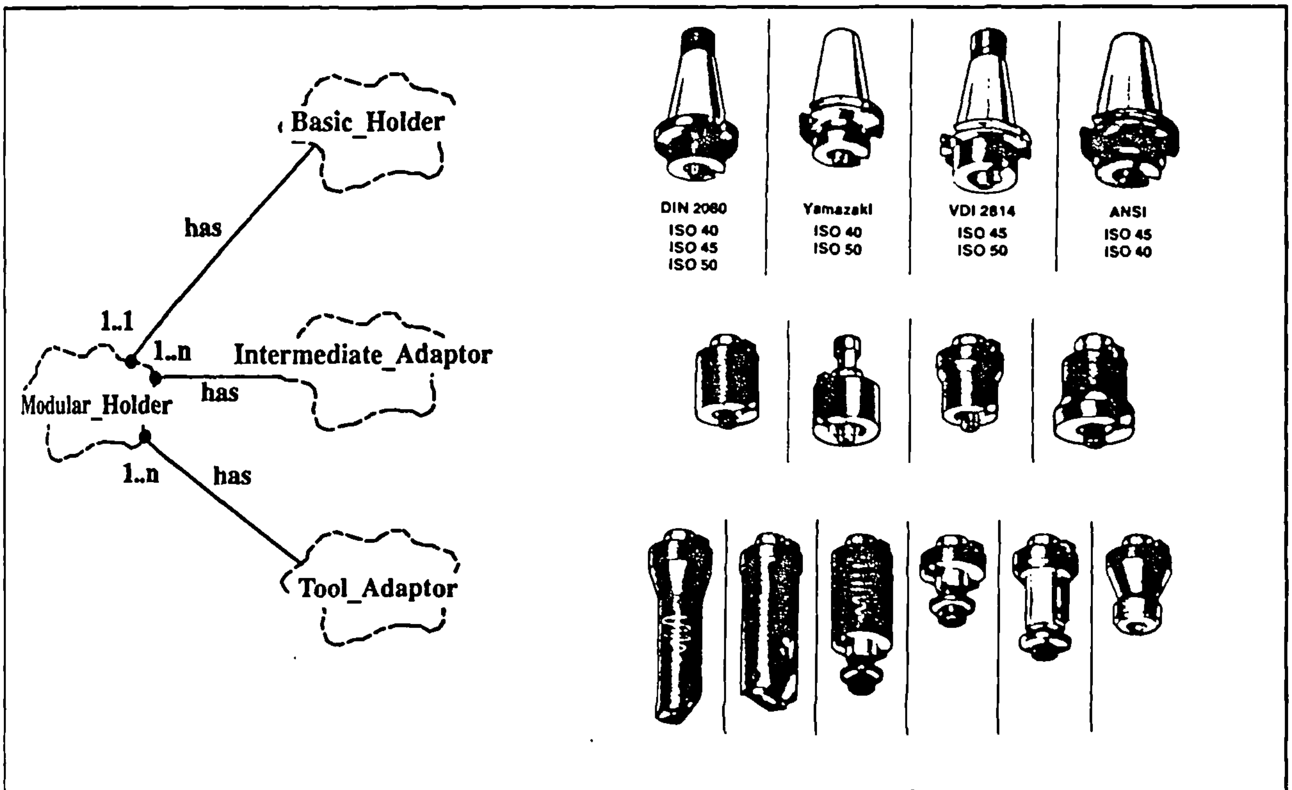
A. Molina | Figure 12.13 Modelling a Contemporary Turning Centre Family | LUT-SE Research Group



A. Molina

Figure 12.14 Modelling Tool Assemblies
(Sandvick Coromant Catalogue C–1100:3–ENG 1991)

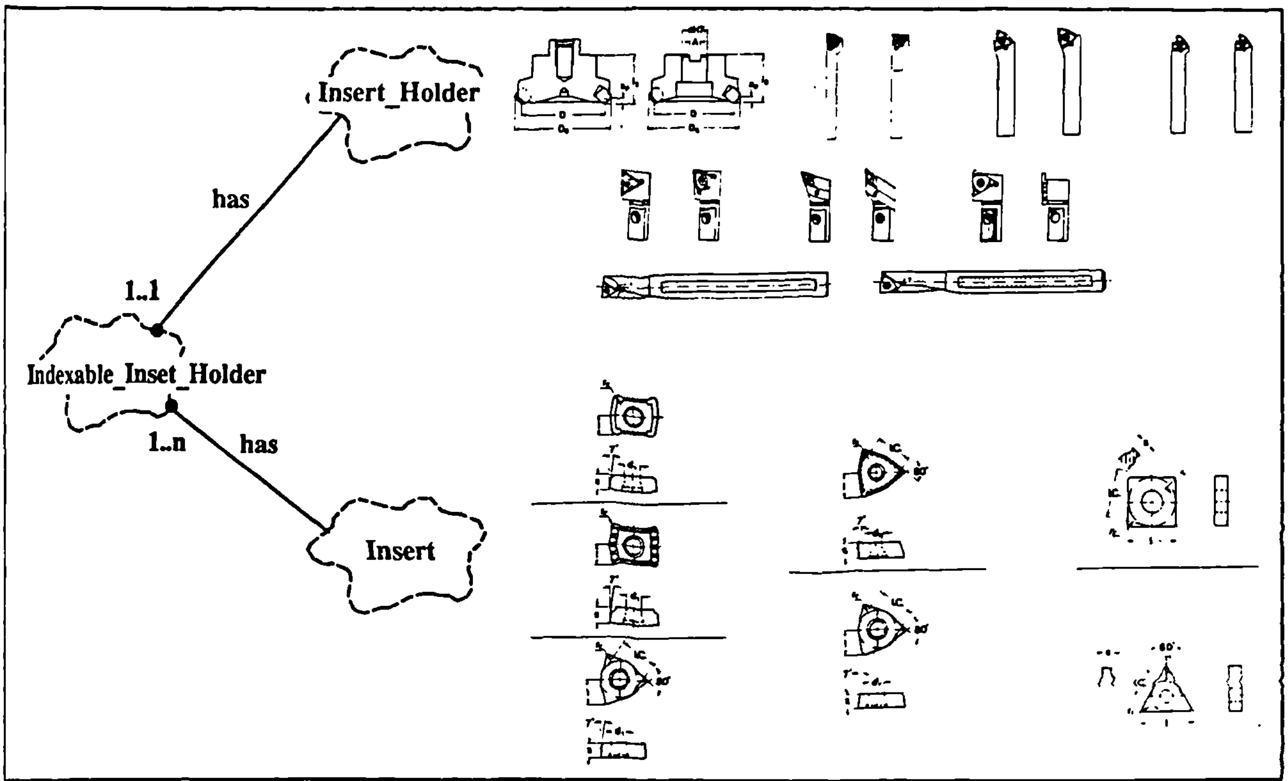
LUT–SE
Research Group



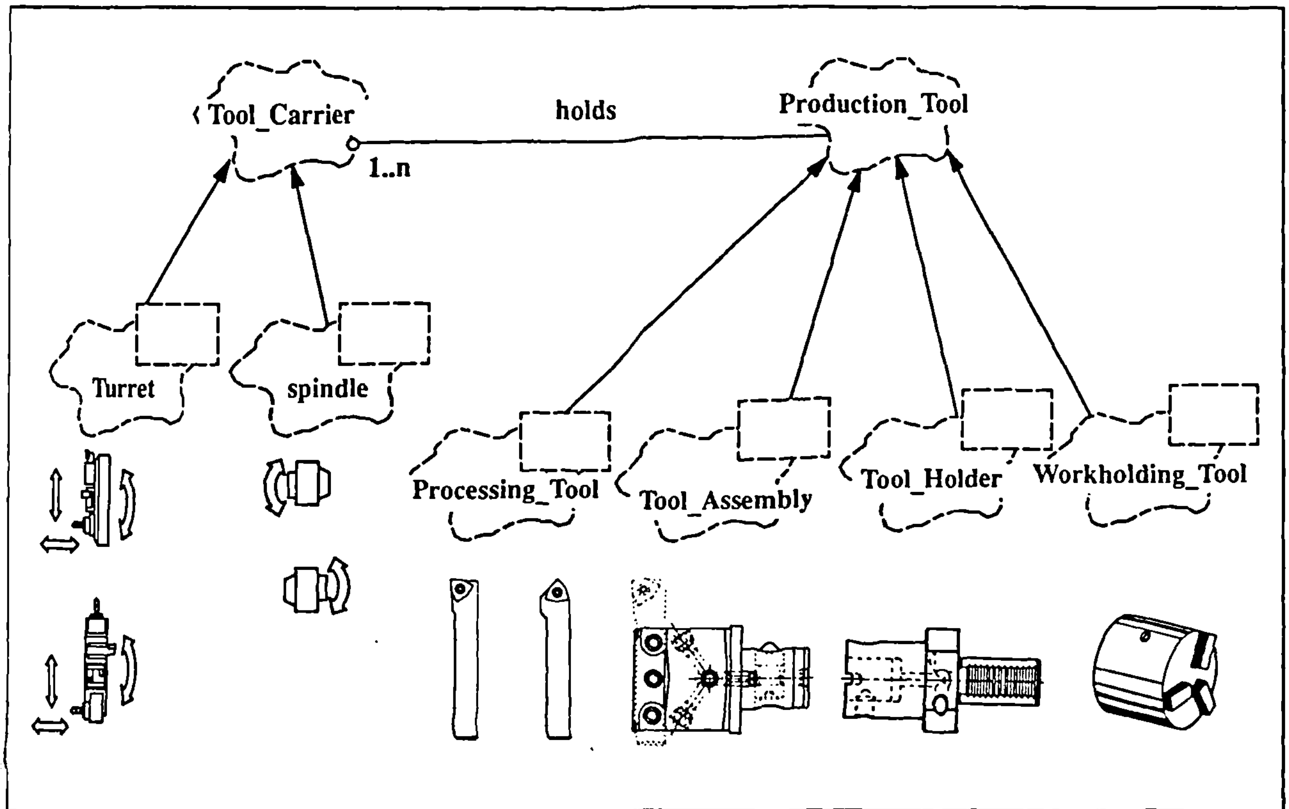
A. Molina

Figure 12.15 Modelling Modular Tooling Holding Systems
(Sandvick Coromant Catalogue C–1100:3–ENG 1991)

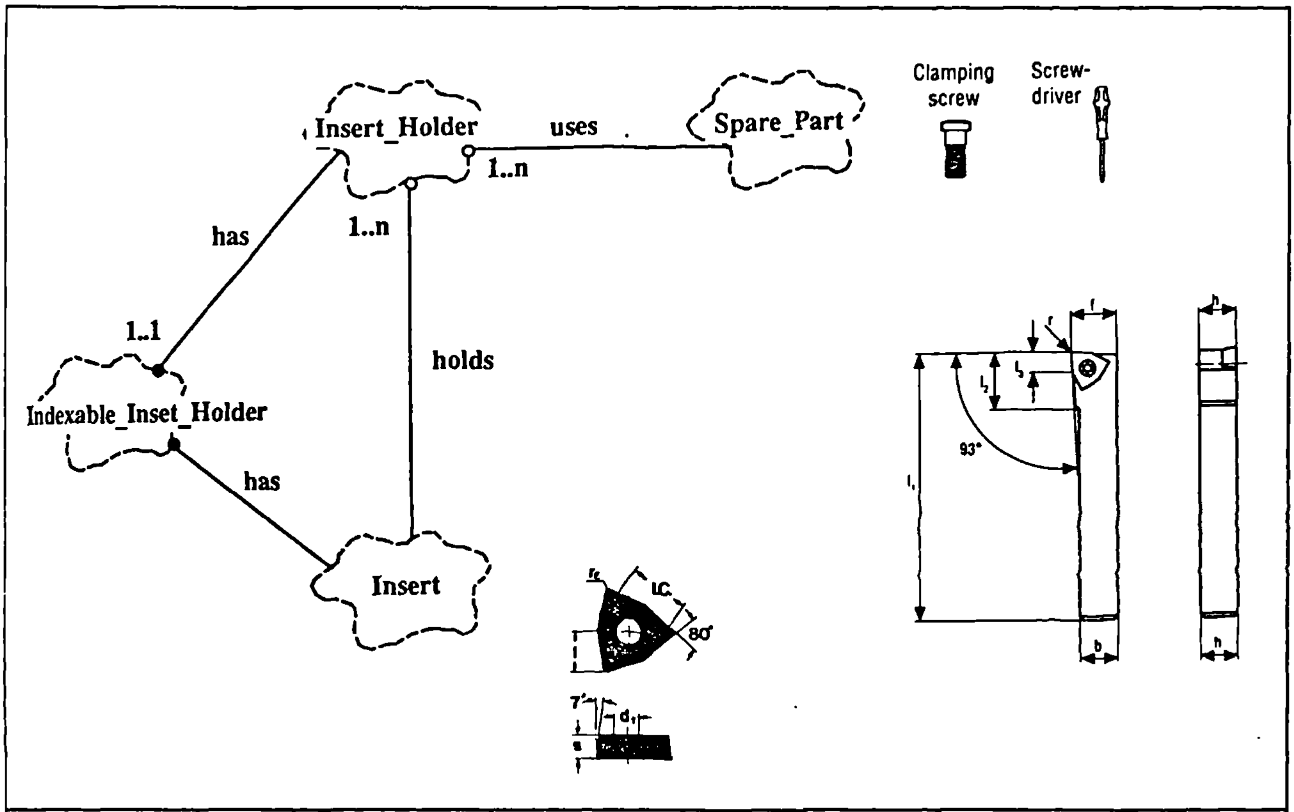
LUT–SE
Research Group



A. Molina	Figure 12.16 Modelling Indexable Insert Holders (Sandvick Coromant Catalogue C–1100:3–ENG 1991 and Komet Catalogue 399 01 015 04–30–03/94)	LUT–SE Research Group



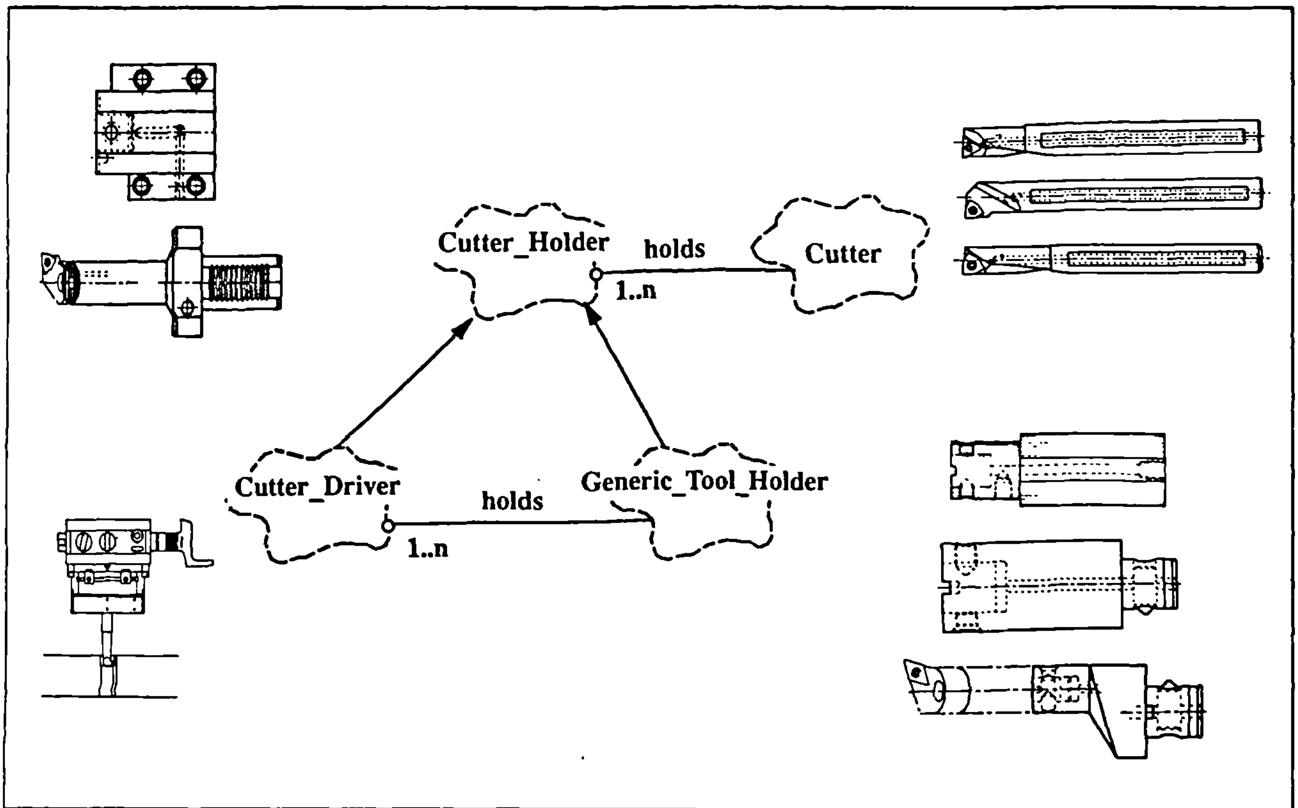
A. Molina	Figure 12.17 Modelling the Tool System Capabilities	LUT–SE Research Group



A. Molina

Figure 12.18 Modelling Indexable Insert Holders

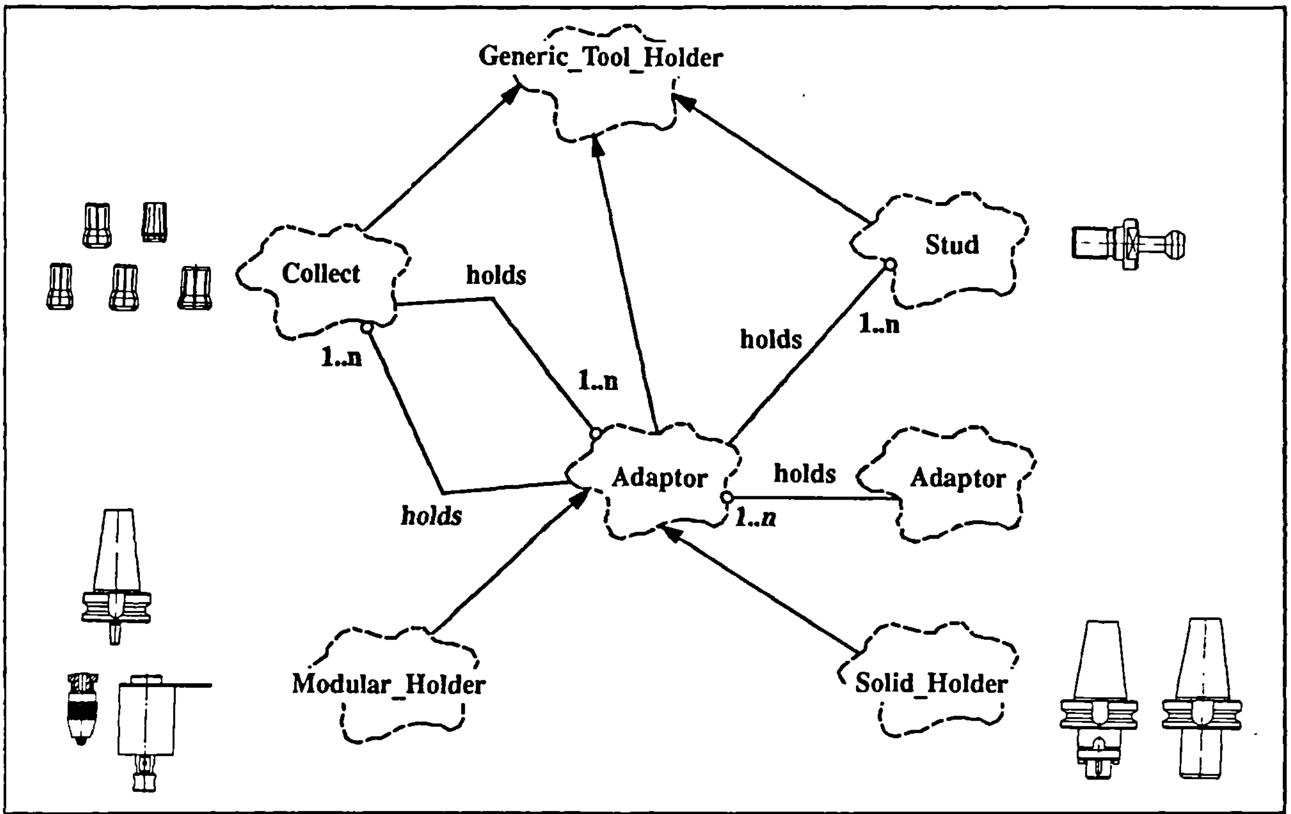
LUT-SE Research Group



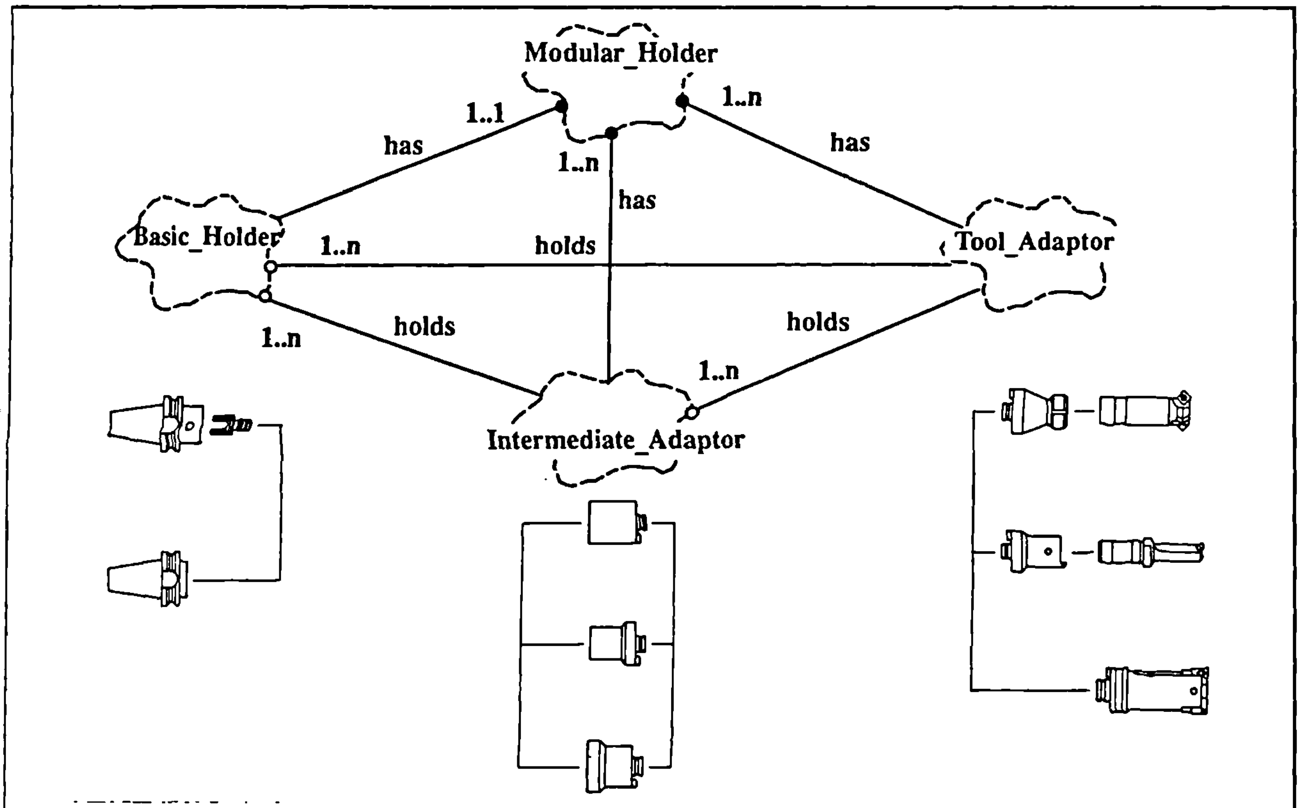
A. Molina

Figure 12.19 Modelling Cutter Holders Capabilities

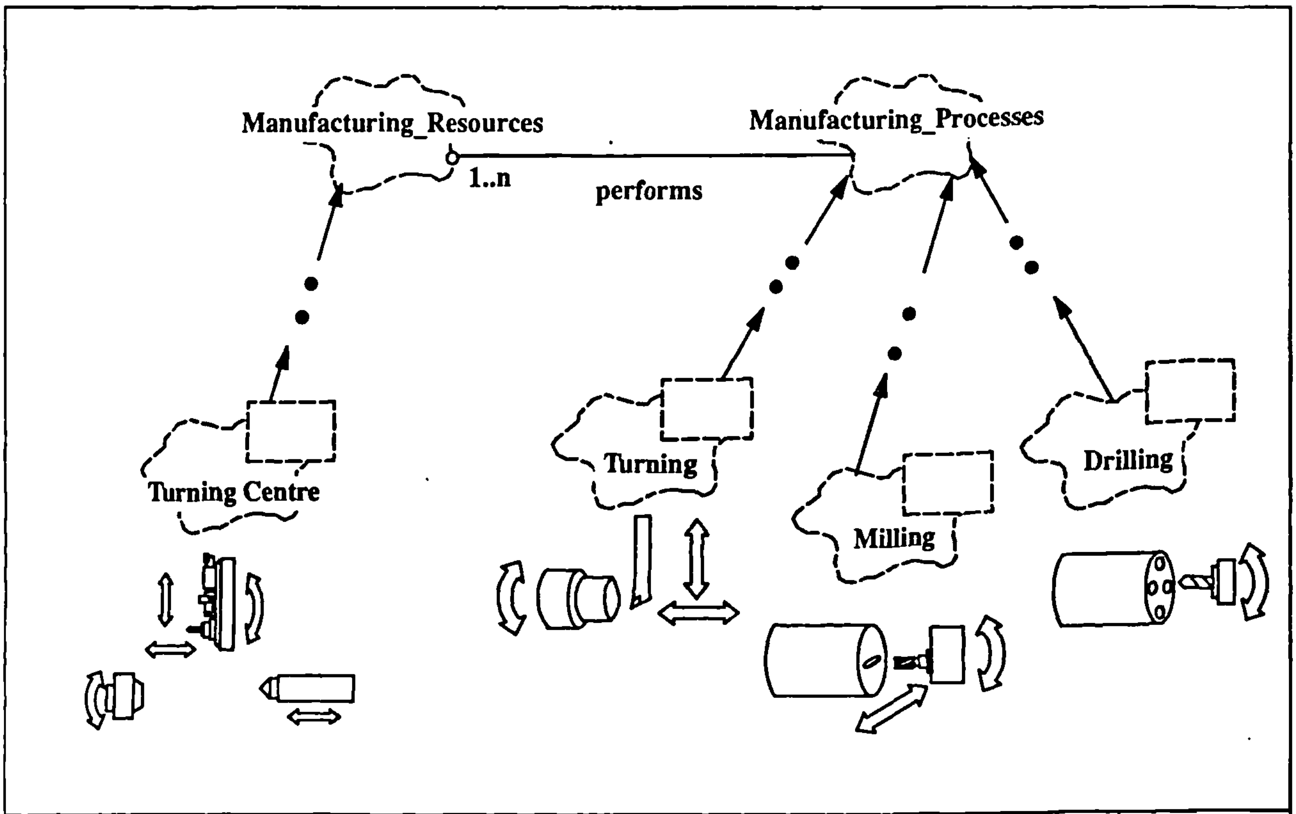
LUT-SE Research Group



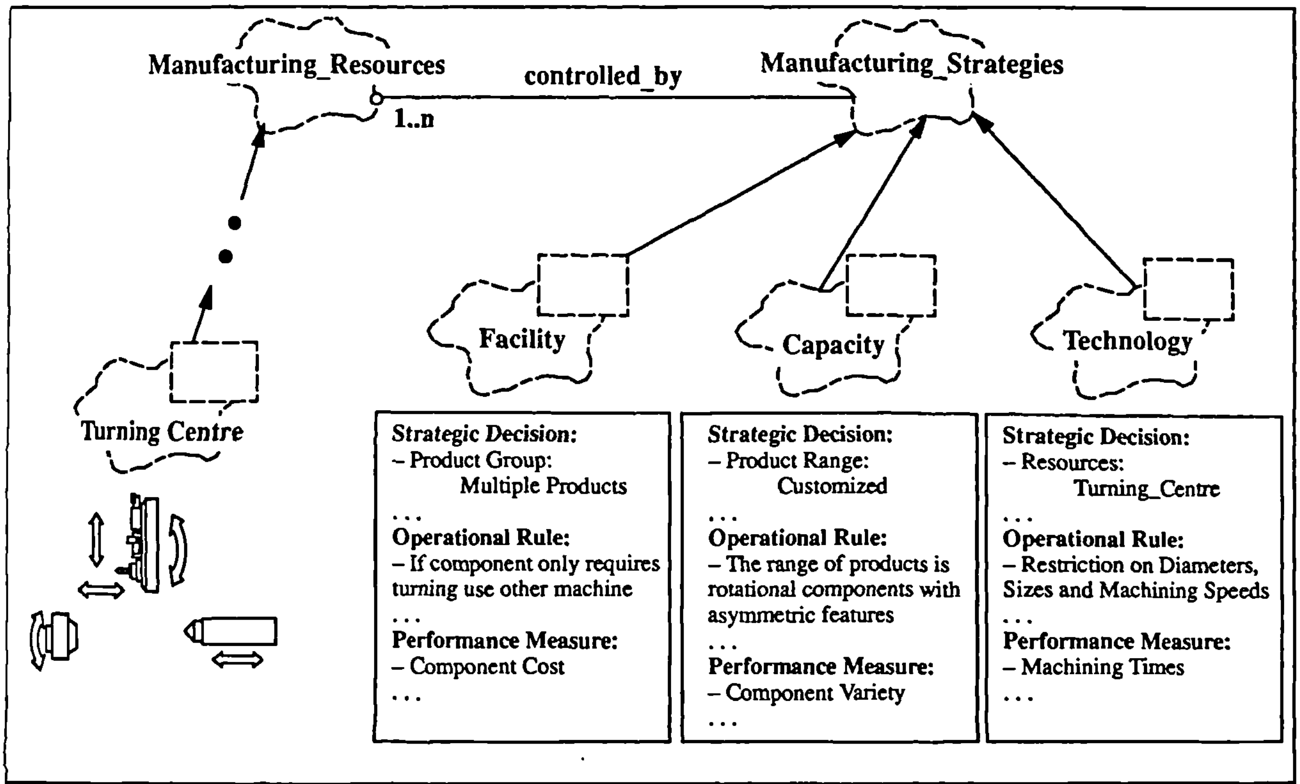
A. Molina	Figure 12.20 Modelling Generic Tool Holder Characteristics	LUT–SE Research Group



A. Molina	Figure 12.21 Modelling Modular Tooling Holding Systems	LUT–SE Research Group



A. Molina | Figure 12.22 Modelling the process capability of a Turning Centre | LUT–SE Research Group



A. Molina | Figure 12.23 Modelling the relation between manufacturing resources and strategies | LUT–SE Research Group

Classification Based on ISO/TC184/SC4/WG8	MANUFACTURING MODEL REPRESENTATION Resource Group: Production Tool			
	TYPE	ADMINISTRATIVE ATTRIBUTES	TECHNICAL ATTRIBUTES	OPERATIONAL ATTRIBUTES
Generic Resource A generic resource is a resources class, for which the characteristics, parameters and assigned construction destinations are completely defined	Cutter	Manufacturers Information = {Name, Address, Telephone} Manufacturers Specification: Manufactures Identification Number Manufacturers Classification Lead Time for Acquisition Acquisition Cost	type of resource = { active, passive } type of cutter = {solid | indexable insert | tool blank} type of mounting = {arbor | shank} type of tool = {boring | drilling | milling | turning} tool geometry = {single point tool | multipoint tool | multirandom edge tool} type of movement = {translation, rotation, stationary} ISO/BSI Code	Tool Company Identification: Tool Identification Number: Tool Company Classification: Average Too Life = { number} Theoretical Tool Life = {number} Date of Acquisition = {Day, Month, Year} Preseting Data = {longitudinal dimension, lateral dimension} Tool Location = {storage room | tool magazine | tool carrier} Tool Status = {available | not available} Tool Alternative: Standard Operation Cost of Tool: £
Specific Resource A specific resource will be built by specification of all parameters that are defined in the related generic resource, except temporate and material planning characteristics	Boring_Tool	Manufacturers Information {Sandvik, Manor Way, 0215504700} Manufacturers Specification: C-1100:3-ENG Manufactures Identification Number: 391.37-16 50 095 Manufacturers Classification: Varilock fine boring head 391.37 Lead Time for Acquisition: 2/3 day Acquisition Cost : £	type of resource = { passive } type of cutter = {indexable insert} type of mounting = {shank} type of tool = {boring} tool geometry = {single point tool} type of movement = {translation, rotation, stationary} Code: BS4193: Part 14 : 1984 S 25R-CTFPR16	
Individual Resource An individual resource is a concrete resource, that individual and temporate characteristics (like material planning dates, wear conditions, cost, etc.) can be assigned to.	Boring_Tool	IDEM	IDEM	Tool Company Identification: Boring Bar Head Tool Identification Number: 210021 Tool Company Classification: S 25R-CTFPR16 Average Too Life: % Theoretical Tool Life: % Preseting Data = {µm, µm} Date of Acquisition: 19/12/93 Tool Location = {tool magazine} Tool Status = {available} Tool Alternative = {Boring Bar} Standard Cost of Tool : £

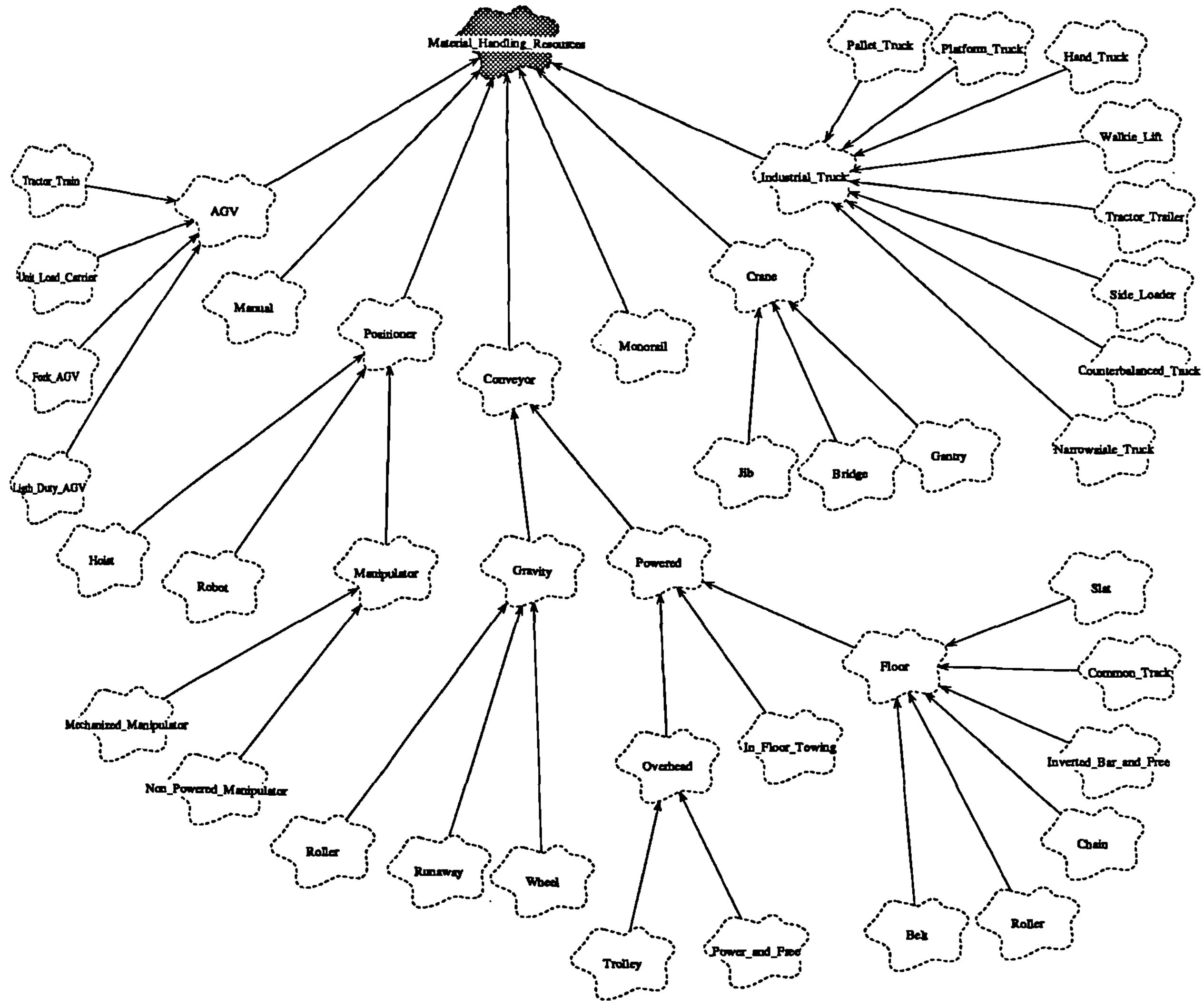
Table 12.1 Administrative, Technical and Operational Attributes of the Resource Group Production Tool and Generic Resource Cutter

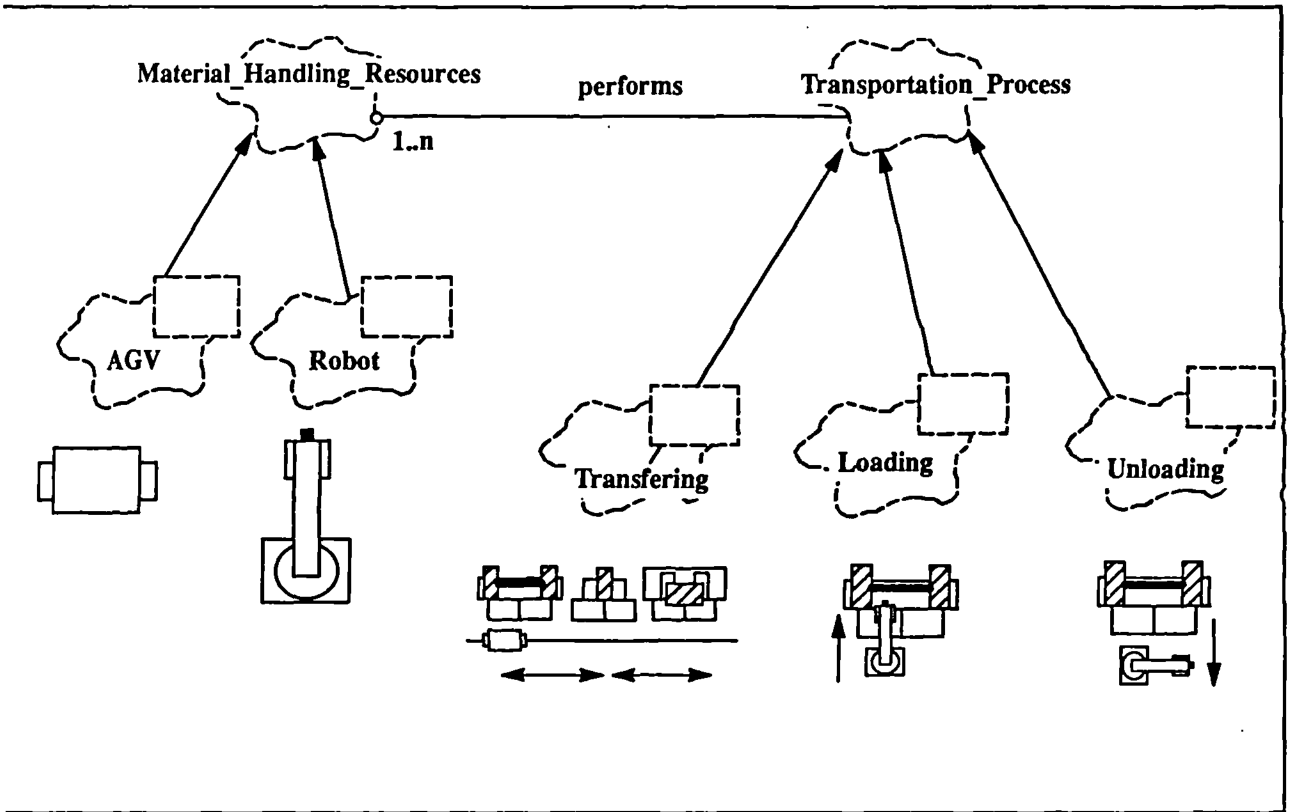
Classification Based on ISO/TC184/SC4/WG8	MANUFACTURING MODEL REPRESENTATION Resource Group: Production Tool			
	TYPE	ADMINISTRATIVE ATTRIBUTES	TECHNICAL ATTRIBUTES	OPERATIONAL ATTRIBUTES

Generic Resource	Cutter
	
	ENTITY Cutter
	ABSTRACT SUPERTYPE OF
	(ONEOF (
	Boring_Tool
	Drill,
	Milling_Cutter,
	Turning_Tool))
	SUBTYPE OF
	(Mass_Reducing_Tool);
	Admin: Administrative_Data
	Technical: Technical_Data;
	Operational: Operational_Data;
	END_ENTITY;
		ENTITY Manufacturers_Information;
		name: STRING(30);
		address: STRING(30);
		phone: INTEGER;
		END_ENTITY;
		
		ENTITY Administrative_Data;
		Manufacturers:
		Manufacturers_Information;
		Manufacturers_Specification:
		STRING(30);
		Manufacturers_Identification:
		STRING(30);
		Manufacturers_Classification:
		STRING(30);
		Lead_Time_for_Acquisition: INTEGER;
		Acquisition_Cost: REAL;
		END_ENTITY;
			TYPE type_of_resource =
			ENUMERATION OF (passive,
			active);
			END_TYPE;
			TYPE type_of_cutter =
			ENUMERATION OF (solid,
			indexable_insert, tool_blank);
			END_TYPE;
			TYPE type_of_mounting =
			ENUMERATION OF (arbor,
			shank);
			END_TYPE;
			TYPE type_of_tool =
			ENUMERATION OF (boring,
			drilling, milling, turning);
			END_TYPE;
			TYPE tool_edge_geometry =
			ENUMERATION OF
			(single_point_tool, multipoint_tool,
			multirandom_edge_tool);
			END_TYPE;
			TYPE type_of_movements =
			ENUMERATION OF (translation,
			rotation, stationary);
			END_TYPE;
			ENTITY BSI_ISO_code; (*...*)
			END_ENTITY;
			
			ENTITY Technical_Data;
			resource_type: type_of_resource;
			cutter_type: type_of_cutter;
			mounting_type: type_of_mounting;
			tool_type: type_of_tool;
			tool_geometry:
			tool_edge_geometry;
			tool_movements: LIST OF
			type_of_movements;
			code: BSI_ISO_code;
			END_ENTITY;
				TYPE location_of_tool = ENUMERATION OF
				(storage_room, tool_magazine, tool_carrier);
				END_TYPE;
				TYPE availability_of_tool = ENUMERATION OF
				(available, not_available);
				END_TYPE;
				
				ENTITY Operational_Data;
				tool_company_identification: STRING;
				tool_identification_number: INTEGER;
				tool_company_classification: INTEGER;
				average_tool_life: INTEGER;
				theoretical_tool_life: INTEGER;
				date_of_acquisition: DATE;
				preseting_data: ARRAY [1:2] OF INTEGER;
				tool_location: location_of_tool;
				tool_availability: availability_of_tool;
				tool_alternative: STRING(30);
				standard_operation_tool_cost: REAL;
				END_ENTITY;

Table 12.2 EXPRESS descriptions of the Administrative, Technical and Operational Attributes of the Resource Group Production Tool and Generic Resource Cutter

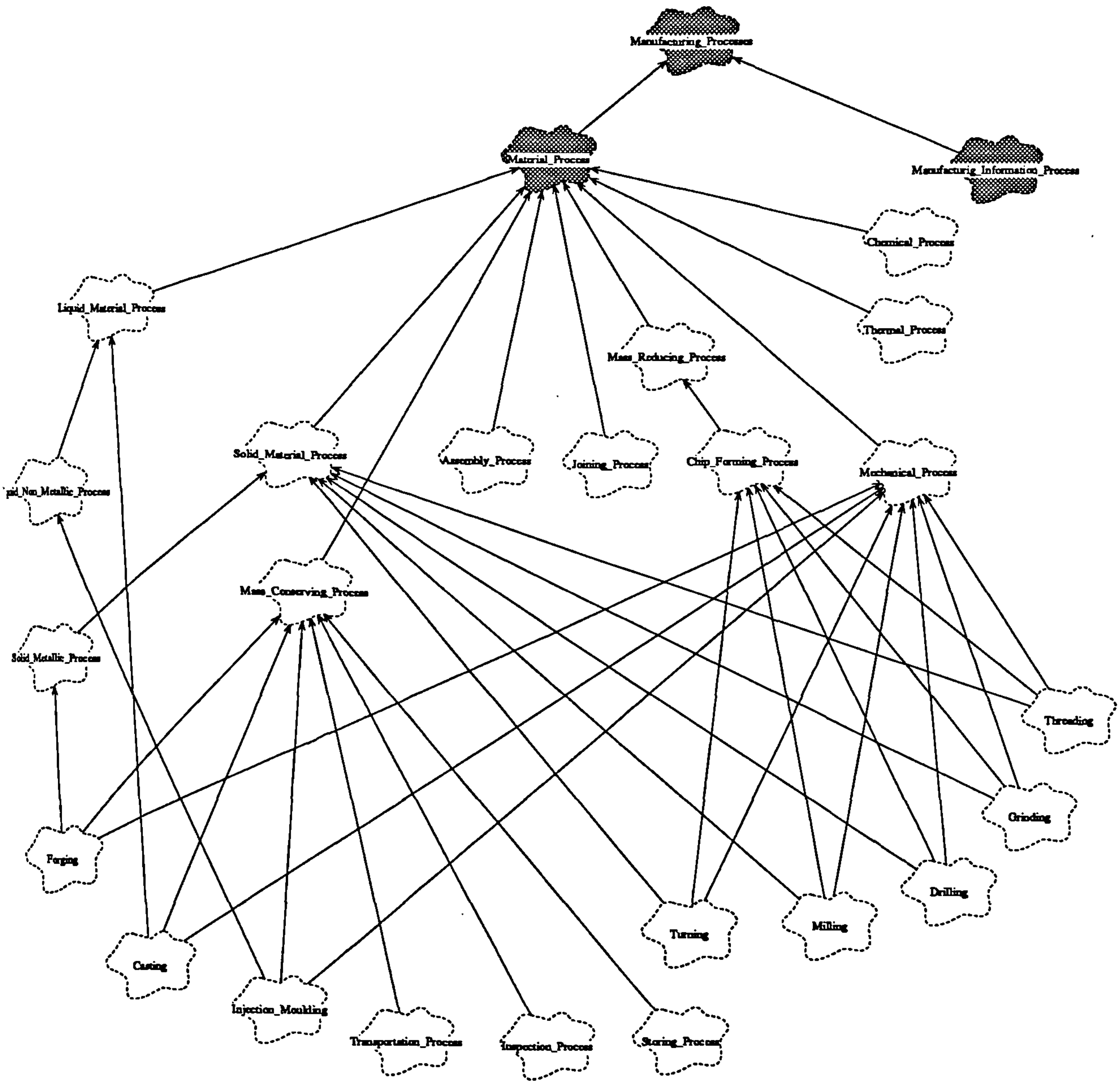
Project Team: LUT-SE Research Group
Title: Figure 12.24 Material Handling Resources Taxonomy
Project: Manufacturing Model
Printed by: A. Molina

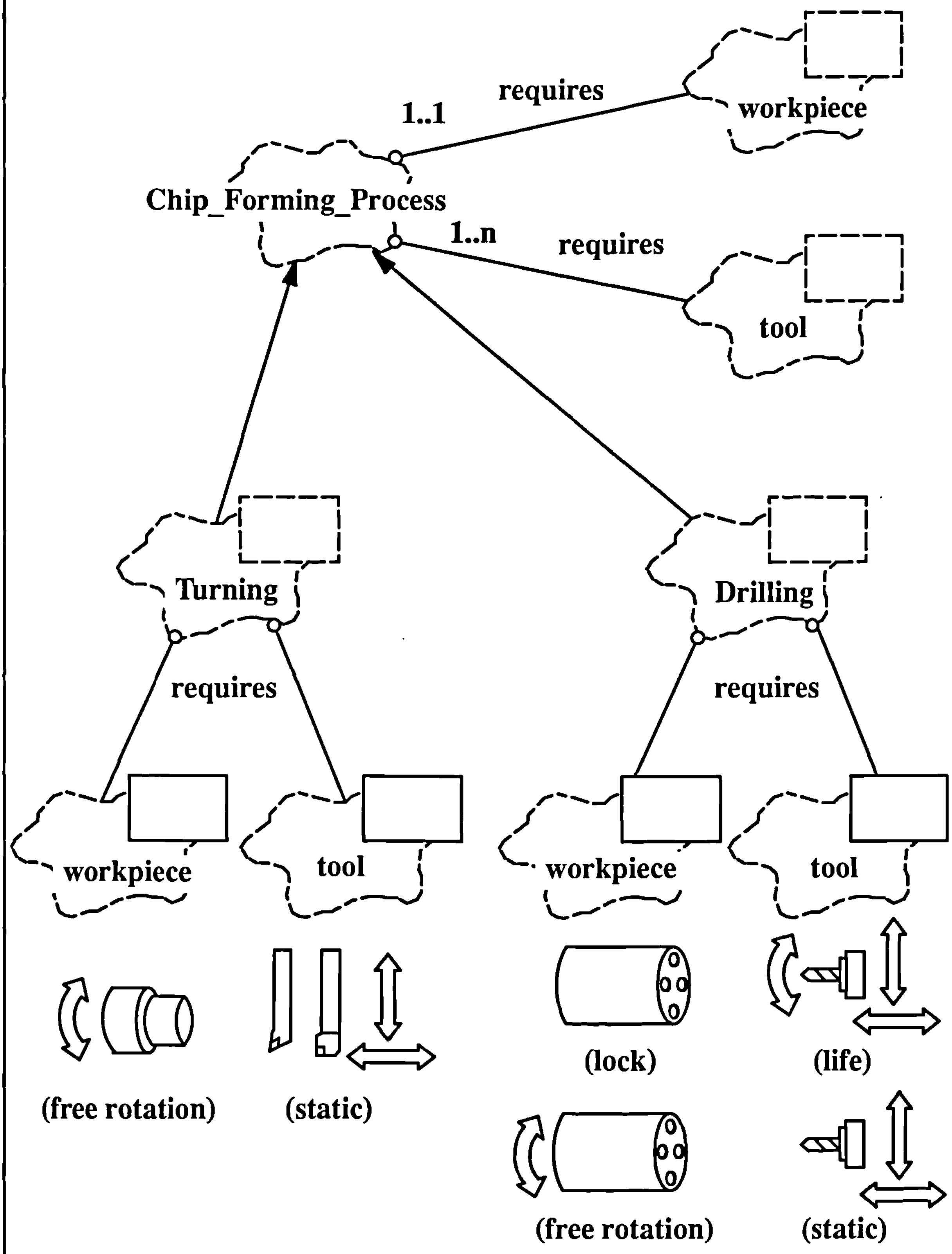




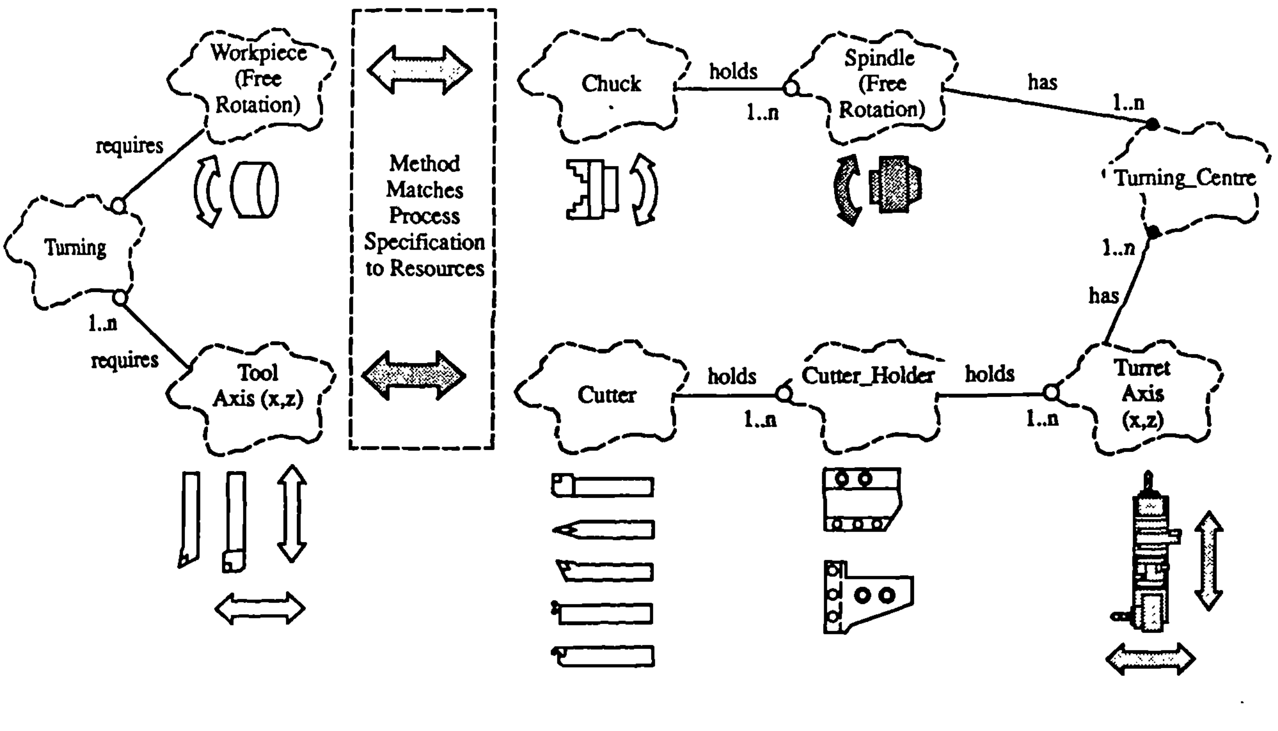
A. Molina	Figure 12.25 Modelling Material Handling Resources Capabilities	LUT–SE Research Group

Project Team:	LUT-SE Research Group
Title:	Figure 12.26 Manufacturing Processes Taxonomy
Project:	Manufacturing Model
Printed by:	A. Molina

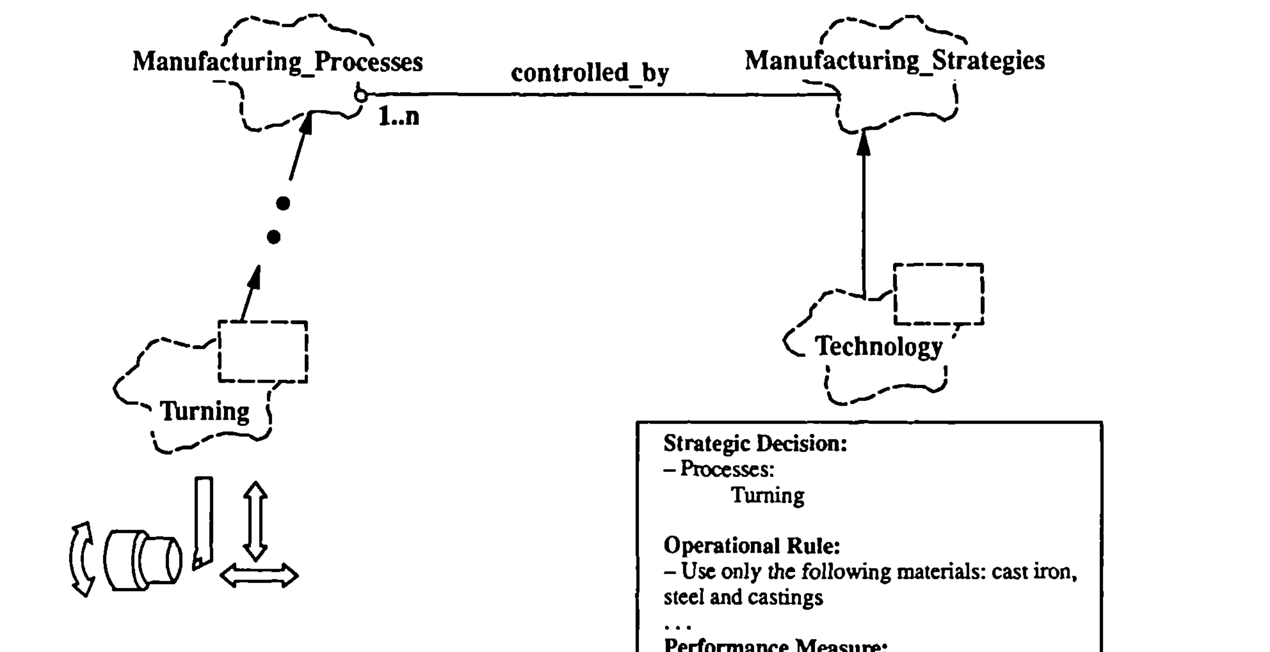




A. Molina	Figure 12.27 Modelling the Machining Process	LUT–SE Research Group



A. Molina	Figure 12.28 Relationships between manufacturing processes and resources	LUT-SE *Research Group*



Strategic Decision:
– Processes:
Turning

Operational Rule:
– Use only the following materials: cast iron, steel and castings
...

Performance Measure:
– Tolerance
– Cost
...

A. Molina	Figure 12.29 Modelling the relation between manufacturing processes and strategies	LUT-SE Research Group

<table>
<tr><th rowspan="2">Classification
Based on ISO/TC184/SC4/WG8</th><th colspan="4">MANUFACTURING MODEL REPRESENTATION
Process Group: Mass Reducing Process</th></tr>
<tr><th>TYPE</th><th>ADMINISTRATIVE ATTRIBUTES</th><th>TECHNICAL ATTRIBUTES</th><th>OPERATIONAL ATTRIBUTES</th></tr>
<tr><td>Generic Process
A generic resource is a resources class, for which the characteristics, parameters and assigned construction destinations are completely defined</td><td>Chip Forming Process</td><td>Subcontractors Information {Name, Address, Telephone}

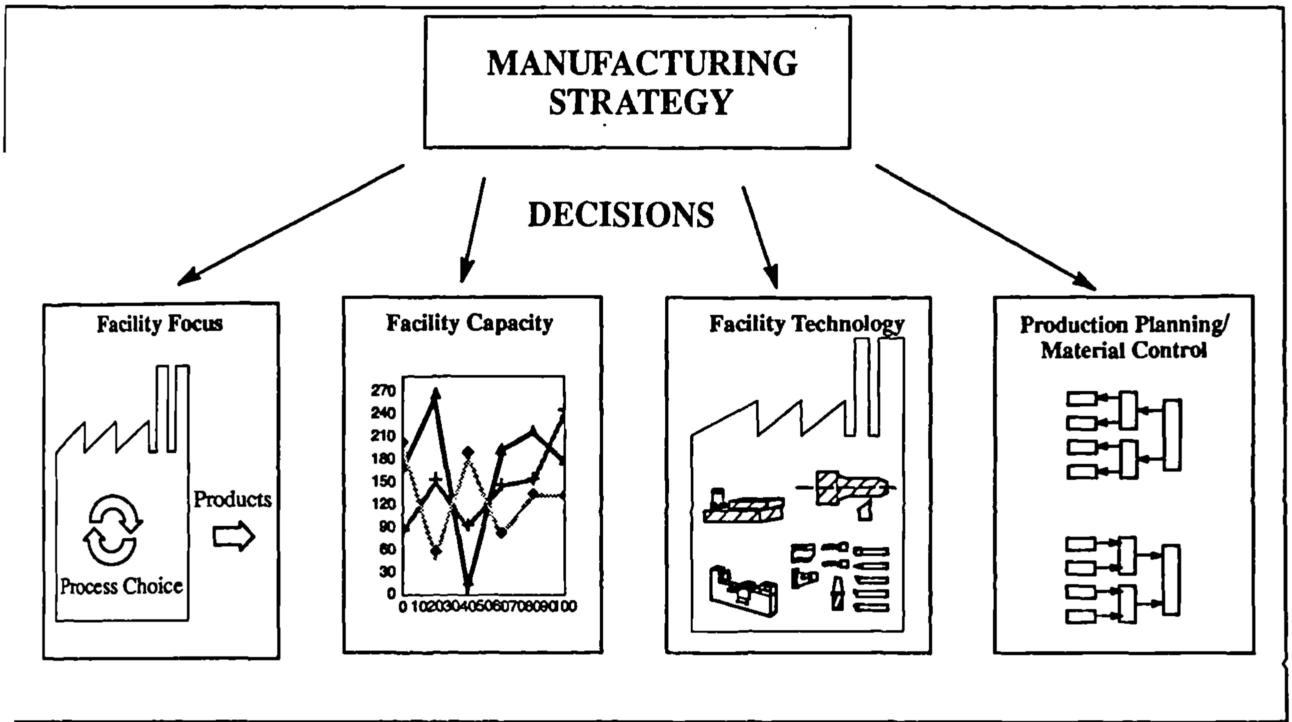
Subcontractors Specifics:
Cost : REAL
Delivery Time: {hours | days | weeks}</td><td>type of process = {Mass Reducing Process}
type of basic process = {mechanical | chemical | thermal}
type of material = {solid, liquid, granular}
workpiece form = {rotational, prismatic, sheet}
tolerance = {min, max};
surface finish = {min, max}
envelope size = {min, max}
minimal economic production volume = number
batch size = {min, max}
lead time initial batch = {high, medium, low}
lead time subs batch = {high, medium, low}
initial cost = {high, medium, low}
production cost = {high, medium, low}
consumables = names
material properties required = {density, hardness, conductivity, youngs_modulus, UTS}</td><td>Process Company Identification:
Process Identification Number:
Process Company Classification:
Process Location = {in-house | subcontract}
Process In-house Location = {station | cell | shop floor}
Process Status = {available | not available}
Process Alternative: name
Standard Operation Cost : £
Standard Operation Time: number</td></tr>
<tr><td>Specific Process
A specific resource will be built by specification of all parameters that are defined in the related generic resource, except temporary and material planning characteristics</td><td>Turning</td><td>Subcontractors Information {HouseTurn, Leicester, 0215504700}
Cost : £
Delivery Time: {days}</td><td>type of process = {Mass Reducing Process}
type of basic process = {mechanical}
type of material = {solid}
workpiece form = {rotational}
....
....</td><td></td></tr>
<tr><td>Individual Process
An individual resource is a concrete resource, that individual and temporary characteristics (like material planning dates, wear conditions, cost, etc.) can be assigned to.</td><td>External Turning</td><td>IDEM</td><td>IDEM</td><td>Process Company Identification: External Turning
Process Identification Number: #22
Process Company Classification: ET#22
Process Location = {station}
Process Status = {available}
Process Alternative: T#24
Standard Operation Cost : £
Standard Operation Time: number</td></tr>
</table>

Table 12.3 Administrative, Technical and Operational Attributes of the Group Mass Reducing Process and Generic Chip Forming Process

Classification Based on ISO/TC184/SC4/WG8	MANUFACTURING MODEL REPRESENTATION Process Group: Mass Reducing Process			
	TYPE	ADMINISTRATIVE ATTRIBUTES	TECHNICAL ATTRIBUTES	OPERATIONAL ATTRIBUTES

Generic Process	Chip_Forming_Process
	
	ENTITY Chip_Forming_Process
	ABSTRACT SUPERTYPE OF
	(ONEOF (
	Boring,
	Drilling,
	Milling,
	Turning))
	SUBTYPE OF
	(Mass_Reducing_Process);
	Admin: Administrative_Data;
	Technical: Technical_Data;
	Operational: Operational_Data;
	requires_workpiece: Workpiece;
	requires_tool: LIST OF Tool;
	END_ENTITY;
	
	ENTITY Turning
	SUBTYPE OF
	(Chip_Forming_Process,
	Mechanical_Process,
	Solid_Material_Process);
	END_ENTITY;
		ENTITY Subcontractors_Information;
		name: STRING(30);
		address: STRING(30);
		phone: INTEGER;
		END_ENTITY;
		
		ENTITY Administrative_Data;
		Subcontractors:
		Subcontractors_Information;
		Cost: REAL;
		Delivery_Time: INTEGER;
		END_ENTITY;
			TYPE typeform = ENUMERATION OF (rotational, prismatic, sheet);
			END_TYPE;
			
			TYPE rating = ENUMERATION OF (high, medium, low);
			END_TYPE;
			
			TYPE properties =
			ENUMERATION OF (density, hardness, conductivity, youngs_modulus, UTS);
			END_TYPE;
			
			ENTITY Technical_Data;
			tolerance_minimum: REAL;
			tolerance_maximum: REAL;
			surface_finish_min: REAL;
			surface_finish_max: REAL;
			form: typeform;
			envelope_max_x: REAL;
			envelope_max_y: REAL;
			envelope_max_z: REAL;
			envelope_min_x: REAL;
			envelope_min_y: REAL;
			envelope_min_z: REAL;
			min_econ_prod_vol: INTEGER;
			min_econ_bat_size: INTEGER;
			max_econ_bat_size: INTEGER;
			lead_time_init_bat: rating;
			lead_time_subs_bat: rating;
			initial_proc_cost: rating;
			production_proc_cost: rating;
			consumables: STRING(120);
			props_required: LIST OF properties;
			END_ENTITY;
				TYPE location_of_process = ENUMERATION OF (in_house, subcontract);
				END_TYPE;
				TYPE in_house_location_of_process =
				ENUMERATION OF (station, cell, shop_floor);
				END_TYPE;
				TYPE availability_of_process = ENUMERATION OF (available, not_available);
				END_TYPE;
				
				ENTITY Operational_Data;
				process_company_identification: STRING(30);
				process_identification_number: STRING(30);
				process_company_classification: STRING(10);
				process_location: location_of_process;
				process_in_house_location:
				in_house_location_of_process;
				process_availability: availability_of_process;
				process_alternative: STRING(30);
				standard_operation_cost: REAL;
				standard_operation_time: REAL;
				END_ENTITY;

Table 12.4 EXPRESS descriptions of the Administrative, Technical and Operational Attributes of the Process Group Mass Reducing Process and Generic Chip Forming Process

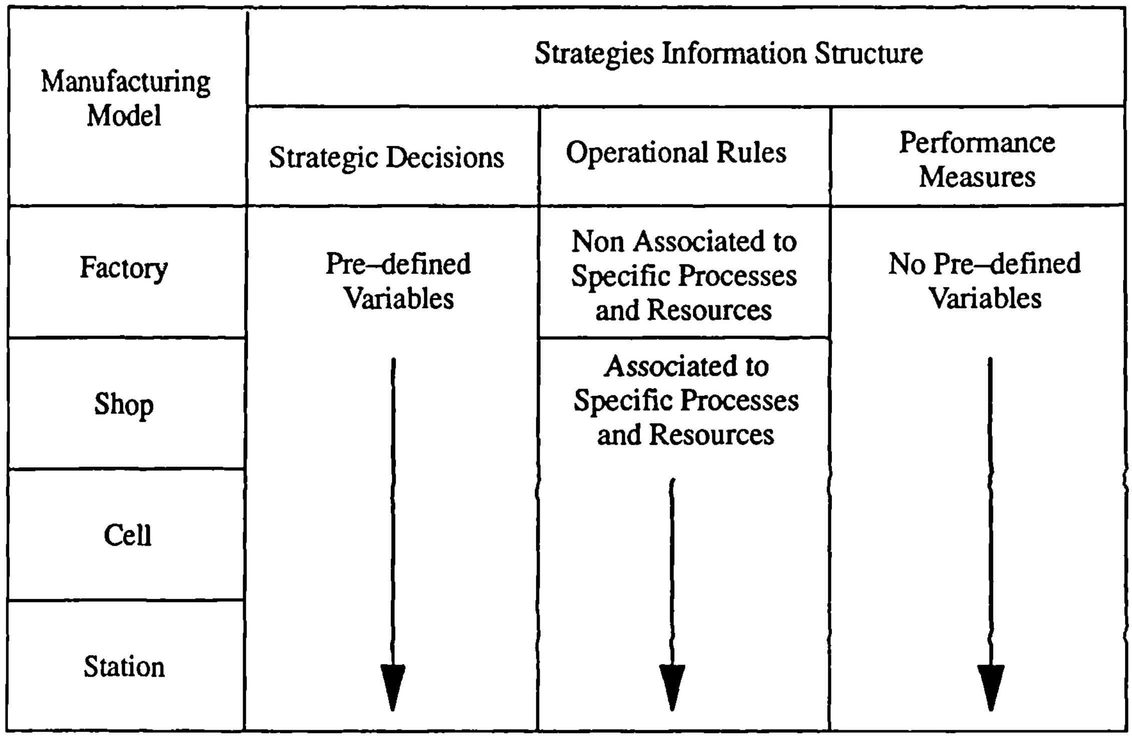


A. Molina | Figure 12.30 Categories of Strategies | LUT-SE Research Group

MANUFACTURING STRATEGIES

STRATEGIC DECISIONS	OPERATIONAL RULES	PERFORMANCE MEASURES
Decisions that define the Facility: **- Focus** **- Capacity** **- Technology** **- Production Planning/ Material Control**	**Rules that enforce decisions:** **- Experience** **- Know-how**	**Performance measures that guide decisions:** **- Cost** **- Quality** **- Flexibility** **- Delivery**

A. Molina | Figure 12.31 Manufacturing Strategies Information Structure: Strategic Decisions, Operational Rules and Performance Measures | LUT-SE Research Group



A. Molina | Figure 12.32 Characteristics of the Manufacturing Strategies at different levels in the Manufacturing Model | LUT–SE *Research Group*

Facility	Strategies Information Structure		
	Strategic Decisions	Operational Rules	Performance Measures
Facility Focus	Product Group: (OKP) One of a Kind Products, (MP) Multiple Products, (FMP) Few Major Products, and (CP) Commodity Products	IF .. THEN Constraints	Cost, Quality Flexibility, Delivery
	Product Volume: (LV–LS) Low Volume – Low Standardization, (LV) Low– to medium Volume, (HV) High– to medium Volume, and (HV–HS) High Volume – High Standardisation	idem	idem
	Product Life Cycle: (NP) New Products, (GP) Growing Products, (MP) Mature Products and (DP) Declining Products	idem	idem
	Production Mode: (P) Project–job shop, (F) Functional–batch, (C) Cellular, (L) Line, and (C) Continuous	idem	idem
Capacity	Product Range: (H) Highly Standard, (S) Standard, (LS) Low Standard and (C) Customized	idem	idem
	Cycle Demand: (HE) Holding Excess Capacity, (HI) Holding Seasonal Inventories, (O) Overtime, and (S) Subcontracting	idem	idem
	Increase Capacity: (O) Overtime, (IS) Increase Shifts, (S) Subcontracting, (EF) Enlarge Facility and (AF) Additional Facility in a new location	idem	idem
Technology	Processes: list of Processes	idem	idem
	Resources: list of Resources	idem	idem
	Configuration: type of facility (HA) Highly Automated, (PA) Partial Automated, (LA) Low Automated and (M) Manual	idem	idem
	Layout: Intracell layouts – (L) Line Layout, (U) U Layout, (S) S Layout, (LL) L Layout, (O) O Layout, (D) D (robot served) Layout and (W) Layout. Intercell layout – (L) Linear Layout, (X) X Layout, (T) T Layout, (N) Network of Cells, (CF) Common Facilities, (CC) Cascading Cells, (PF) Parallel Flow–lines, and (VC) Virtual Cells.	idem	idem
Production Planning/ Material Control	Master Scheduling: (ATO) Assemble to Order, (DTO) Design to Order, (ETO) Engineered to Order, (MTO) Make to Order, and (MTS) Make to Stock.	idem	idem
	Shop Floor Control: Push type and Pull type	idem	idem
	Material Planning: Time–phased and Rate–based	idem	idem
	Facility Scheduling: no predefined values	idem	idem

Table 12.5 Information Structure of the Manufacturing Strategies

Classification Based on ISO/TC184/SC4/WG8	MANUFACTURING MODEL REPRESENTATION Process Group: Strategies			
	TYPE	ADMINISTRATIVE ATTRIBUTES	TECHNICAL ATTRIBUTES	OPERATIONAL ATTRIBUTES

Generic Strategies	Facility_Strategies
	
	ENTITY Facility_Strategies
	ABSTRACT SUPERTYPE OF
	(ONEOF (
	Product_Group,
	Product_Volume,
	Product_Life_Cycle,
	Production_Mode))
	SUBTYPE OF (Strategies);
	Admin: Administrative_Data;
	Technical: Technical_Data;
	END_ENTITY;
		ENTITY Responsible_Information;
		name: STRING(30);
		position: STRING(30);
		phone INTEGER:
		END_ENTITY;
		
		ENTITY Administrative_Data
		Responsible: Responsible_Information;
		Date_of_Issue: date;
		END_ENTITY;
			TYPE Product_Group_Values =
			ENUMERATION OF (OKP, MP, FMP, CP);
			END_TYPE;
			
			TYPE Product_Volume_Values =
			ENUMERATION OF (LV_LS, LV, HV, HV_HS);
			END_TYPE;
			
			TYPE Product_Life_Values =
			ENUMERATION OF (NP, GP, MP, DP);
			END_TYPE;
			
			TYPE Production_Mode_Values =
			ENUMERATION OF (P, F, C, T, L, C);
			END_TYPE;
			
			TYPE Facility_Decision = SELECT
			(Product_Group_Values,
			Product_Volume_Values,
			Product_Life_Values,
			Production_Mode_Values);
			END_TYPE;
			
			ENTITY Operational_Rules;
			text: STRING (200);
			END_ENTITY;
			
			ENTITY Performance_Measures;
			text: STRING (30);
			END_ENTITY;
			
			ENTITY Technical_Data;
			Value: LIST OF Facility_Decision;
			Rules: LIST OF Operational_Rules;
			Measures: LIST OF Performance_Measures;
			END_ENTITY;
				NOT DEFINED

Table 12.6 EXPRESS descriptions of the Administrative and Technical Attributes of the Group Strategies and Generic Class Facility Strategies

Chapter 13

Case Study: Modelling a Contemporary Turning Centre at the Station Level

13.1. Introduction

In order to test the concept of the Manufacturing Model created by the author, a case of study has been pursued and the results are presented in this chapter. The case study is based on the information collected from public domain pamphlets of the Yamazaki Company and a visit to the Worcester Factory. The objective is to model a real facility and its capabilities, in this case a Rotational Part Station of the Worcester Factory, and in particular, the ST 40N ATC Mill Center.

13.2. Modelling the Rotational Parts Station Capability

Three brochures were employed to gather the information required to model the Rotational Parts Station of the Worcester Factory: "Mazak European Factory", "Slant Turn 40N ATC Mill Center" (ST–40N ATC M/C.86.1) and "ATC Mill Centers" (ATC M/C T32 90.6). Figure 13.1 shows the principal elements of the Rotational Parts Station described based on the generic classes of resources, processes and strategies presented in Chapter 12. The following subsections explain in detail how these generic classes have been used to represent the capabilities of the Rotational Parts Line and the Slant Turn 40N ATC Mill Center. The detailed development of this case study is described in appendix G.

13.2.1. Modelling the Turning Centre Capabilities

The class turning centre was selected to represent the Slant Turn 40N ATC Mill Center (see Chapter 12, Subsection 12.4.1, figure 12.13). The turning centre generic class is an specialization of the machine tool class. Therefore the machine tool is composed of machine tool components. The Slant Turn 40N ATC Mill Center has been defined as an instance of the generic class turning centre (Figure 13.2). The important components of the Slant Turn 40N ATC Mill Center are: turret, spindle, CNC and tool magazine. These component allows the representation of the machine capabilities. The representation of the turret and spindle describe the feasible movements and determine the machining space. The CNC defines the CNC language capabilities and controlled axis. FInally, the tool magazine describes the tooling capabilities and how many different tools can be used without outside change of tool.

13.2.2. Modelling the Tooling System

The tooling system of the Slant Turn 40N ATC Mill Center shown in figure 13.3 can be described using the generic classes of Production Tools (see Chapter 12, Subsection 12.4.1, figures 12.14 through 12.21). Because these generic classes allows the representation of a wide range of cutters and holders, the modelling work is simplified and requires only to chose the suitable generic class to represent a particular tool. For example, figure 13.4 illustrates the life tooling system using the generic classes Generic Tool Holder and Cutter. In figure 13.5 the static tooling system is modelled using the generic classes of Cutter Holder, Generic Tool Holder and Cutter.

13.2.3. Modelling the Turning Centre Process Capability

The definition of the process capability of the Slant Turn 40N ATC Mill Center is done by defining the set of processes that this machine can perform. The manufacturing process generic class, and in particular the chip forming process class, are employed to model this capability (see Chapter 12, Section 12.5, figure 12.27). The chip forming process definition is based on the specification of the tool to be used and the type of workpiece, together with their respective movements along the axis. The use of the semantic relation *performs* allows the representation of the Slant Turn 40N ATC Mill

Center process capabilities by associating this machine with the different type of process it can perform (Figure 13.6).

13.2.4. Modelling the Rotational Parts Station Configuration

The Station Level of the Manufacturing Model is employed to model the Rotational Parts Station configuration. This configuration has the following components: the Slant Turn 40N ATC Mill Center, the Mazak Loading Robot , 2–Pallet Changer and a set of controllers (i.e. Cell Controller, Robot Controller, AJC Controller). However in this case study, in addition to the Slant Turn 40N ATC Mill Center, only the Mazak Loading Robot and 2–Pallet Changer were modelled. The figure 13.7 shows how these major components of the Rotational Parts Station are represented.

13.2.5. Modelling the Rotational Parts Station Strategies

Strategies in the Manufacturing Model represent decisions made on the organisation and the use of the resources (Chapter 12, Section 12.6). This important aspect has been modelled in this case study in order to represent strategic decisions and operational rules which govern the Rotational Parts Station. As explained in Chapter 12, Section 12.6, the novel approach of representing strategies allows the model to reflect the company choices on how to operate its facility in order to achieve the manufacturing objectives. The strategies where classified into four major categories, in this case study only the following three categories have been used: facilities, capacities and technology. Each category was subdivided into strategic decision variables and for each of these variables a set of operation rules and performance measures were defined. Tables 13.1, 13.2 and 13.3 represent the structure and content of the strategies related to the Rotational Parts Station.

The strategies related to the facility focus are: Product Group, Product Volume, Product Life Cycle and Production Mode. The Rotational Parts Station is dedicated to the production of multiple components, where the components require different machining processes beside the one of turning, therefore this station can not be used on components where only turning is required. The station performance is measured by analysing the variety of components which can be machined at a given time. Consequently the strategic decision variable Product Group has been defined to be Multiple Products (Table 13.1). Because the batch sizes can be of one component to n components the Product Volume was described to be Low Volume –Low Standardisation. As the cost per part should be kept fix regardless the batch size, this is the most important performance measure. The station is dedicated to products which could be at four different stages of their life cycle, i.e. introduction (new products), growth, maturity, and decline. However the station is mainly machining new components and maturity components (Machine Tool Parts). The characteristic of the new components is that they should be rotational with asymmetrical features, and their cost should be competitive. On the other hand, parts belonging to different machine models are machined in this station. The most important rule is to give priority to produce the type of parts required to meet production lead times. Finally the Production Mode has been described to be Batch, Cell and Group Technology (GT). Even when there is no standard GT method, each Rotational Station has different tool configuration, therefore the range of products each of them can machine varies. The rule of thumb employed by the company is to assign the parts to the machine which can produce the part in the minimum set of operations and set–ups. Table 13.1 summarizes the strategic decisions, operational rules and performance measures related to facility strategies.

The strategic decisions related to capacity strategies have been classified in: Product Range, Cycle Demand and Increase Capacity. New components which are rotational with asymmetric features are evaluated and accepted if they are produced cost/effective. The Product Range variable has been defined as Customized with a tendency to increase product flexibility by incrementing the number of products that can be produced at the station. Performance measures for these strategies are: cost per part, machining times, components variety, delivery times, meeting of deadlines, and costs and due dates of subcontractors. The capacity strategies are presented in Table 13.2.

The last type of strategies associated with the Rotational Parts Station are the most common to all the industrialist: Technology strategies. These strategies have been grouped in four categories: Processes, Resources, Configuration and Layout. The first two are related to how the processes and resources are utilized, what restrictions are imposed on them, and what operational rules (rules of thumb) are employed. Table 13.3 illustrates the following Technology Strategies associated to processes: criteria of choice (e.g. minimize cost, minimize lead time, maximize quality), type of material to be used (e.g. cast iron, steel and castings) and the restriction imposed on the use of the Slant Turn 40N ATC Mill Center of not machining components that only require turning. For each different resource, which belongs to the station, restrictions or rules can be applied. For example, the maximum machining diameter of the Slant Turn 40N ATC Mill Center is constrained by the specification of the work handling robot. In Table 13.3 a list of the operational rules which are applied to the resources is presented. The configuration variable specifies the important decisions made related to the configuration such as the use of CNC, Robot for Loading/Unloading, Automatic Tool Changer, Local Buffer and integration with DNC and multiple FMCs. Finally the variable related to Layout has been defined as a special purpose layout for rotational components using a D Robot Served Type Layout.

13.3. The instantiation of the Rotational Parts Station Model

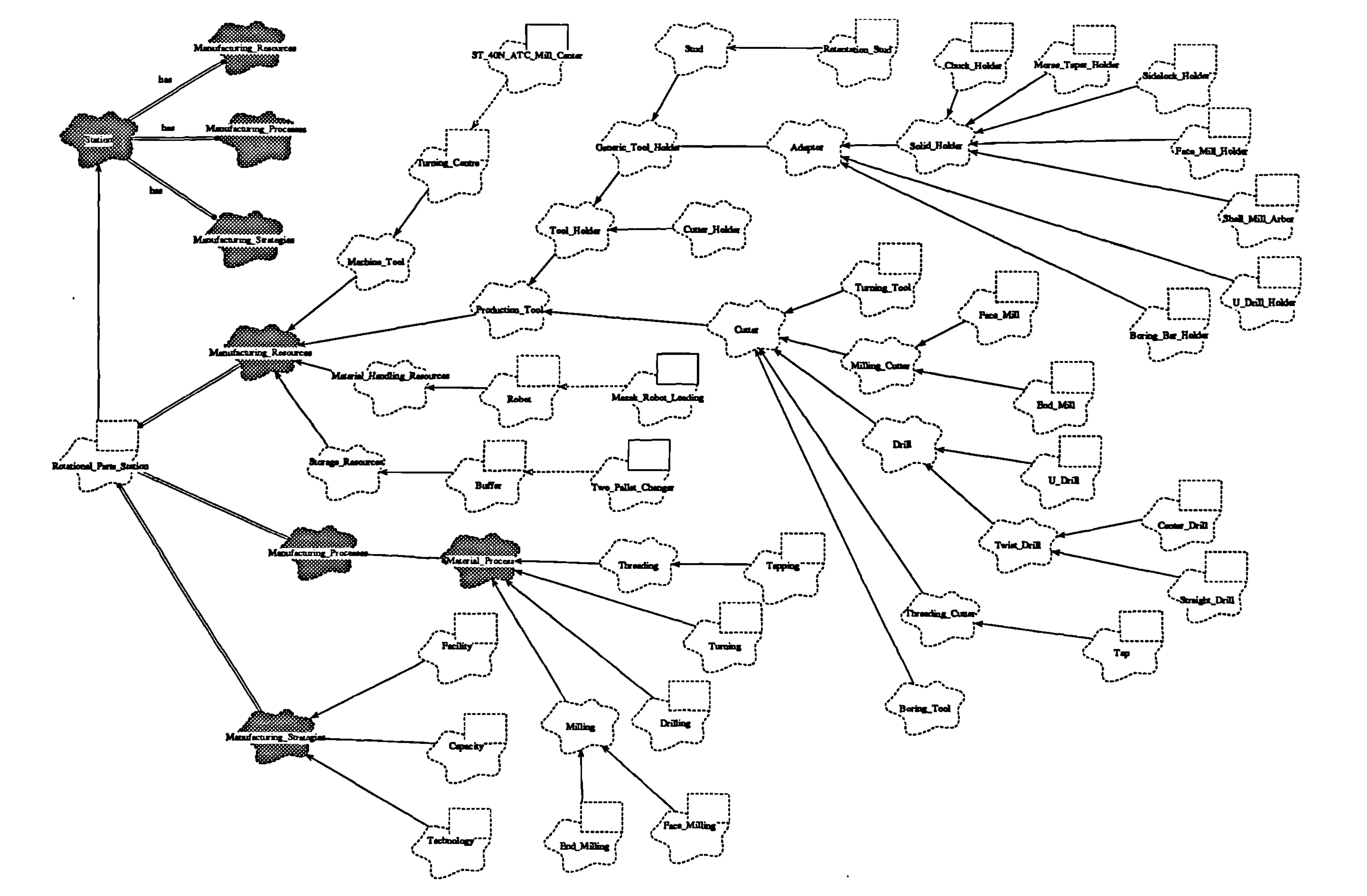
Once the model has been completed, it has to be instantiated to be usable for the data–driven applications. Therefore all the classes in the Rotational Parts Station model have been instantiated and are presented in appendix H. The model has been populated using the Manufacturing Model prototype described in appendix J. The Manufacturing Model Manager of this prototype facilitates the creation of the instances for each of the classes defined in the model. The instantiation is a straightforward process using a set of menus offered by the Manufacturing Model Manager to the user. The creation of all these instances generates the Manufacturing Model database. This database can be browsed using the Manufacturing Model Interfaces described in appendix J. This Manufacturing Model database is ready to be used by data–driven applications.

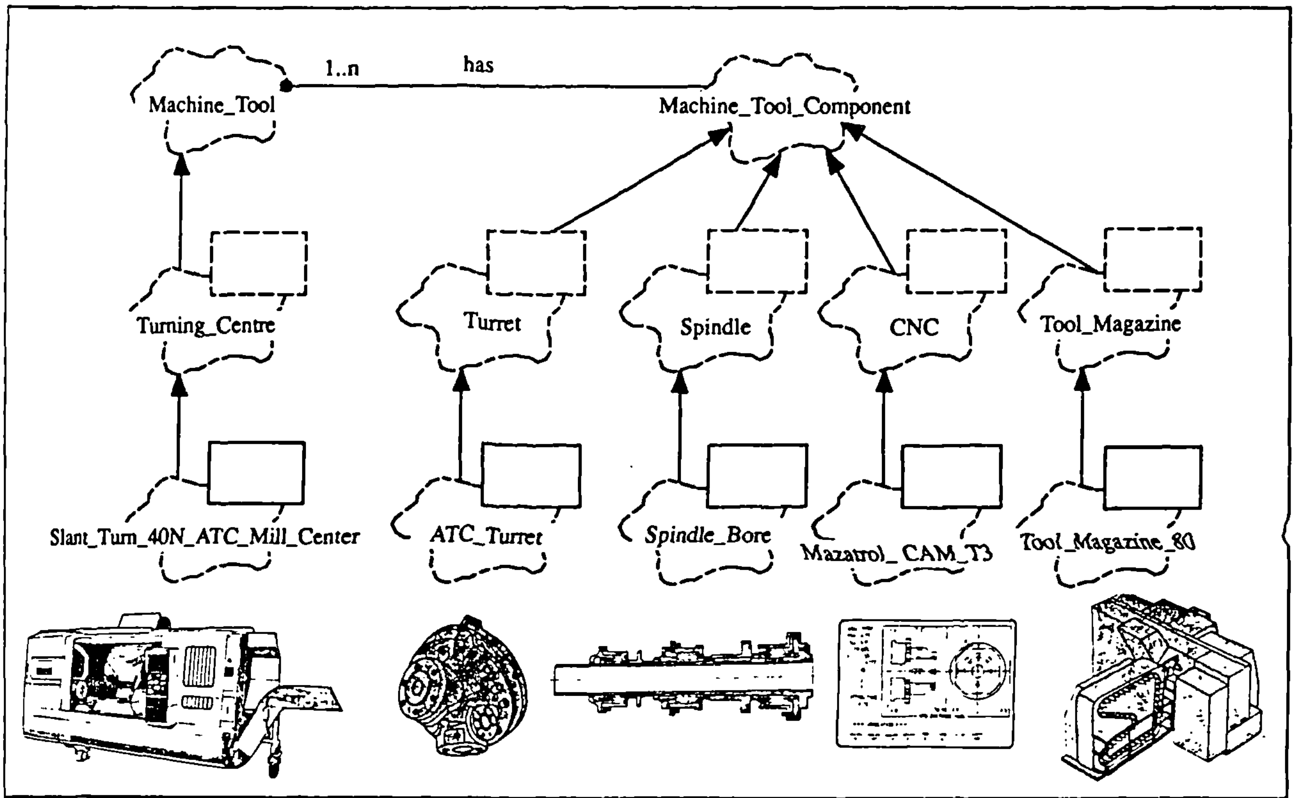
13.4. The applicability of the Station Level of the Manufacturing Model

Two main uses of the Station Level of the Manufacturing Model have been foreseen by the author in the near future. The first is the research within the MOSES project. The work carried out by Mr. T.I.A. Ellis in the area of Design for Manufacture will use the Station Level of the Manufacturing Model to capture and describe the *manufacturing capability* of a contemporary turning centre used by one of the industrial collaborators to produce rotational components (Ellis 1995).

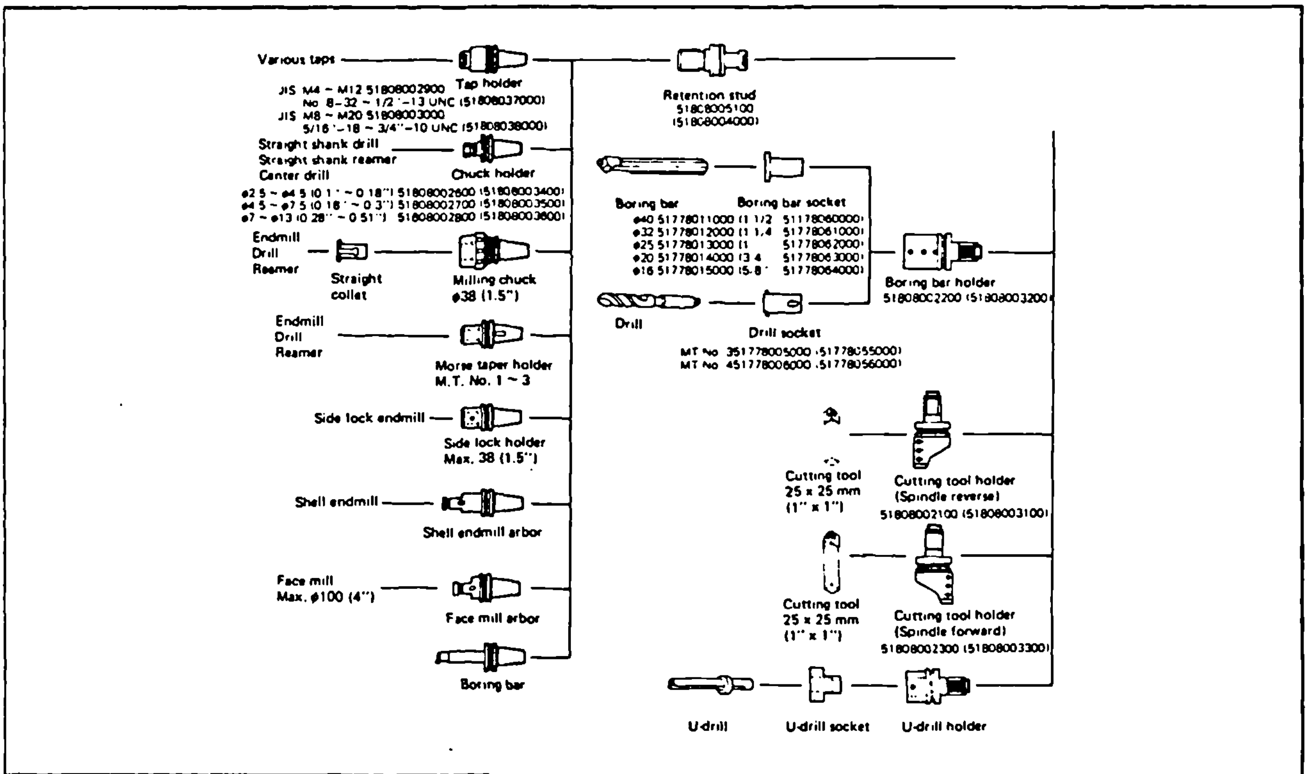
The close contact with the industry has shown a second more restricted, but valuable usage of the Manufacturing Model as the underpinning structure to develop efficient and readily implemented databases for contemporary company use. Such databases will constitute a standardised source of key manufacturing information, i.e. machine tools, production tools, and manufacturing processes.

Project Team:	LUT-SE Research Group
Title:	Figure 13.1 Rotational Parts Station Model
Project:	Manufacturing Model
Printed by:	A. Molina

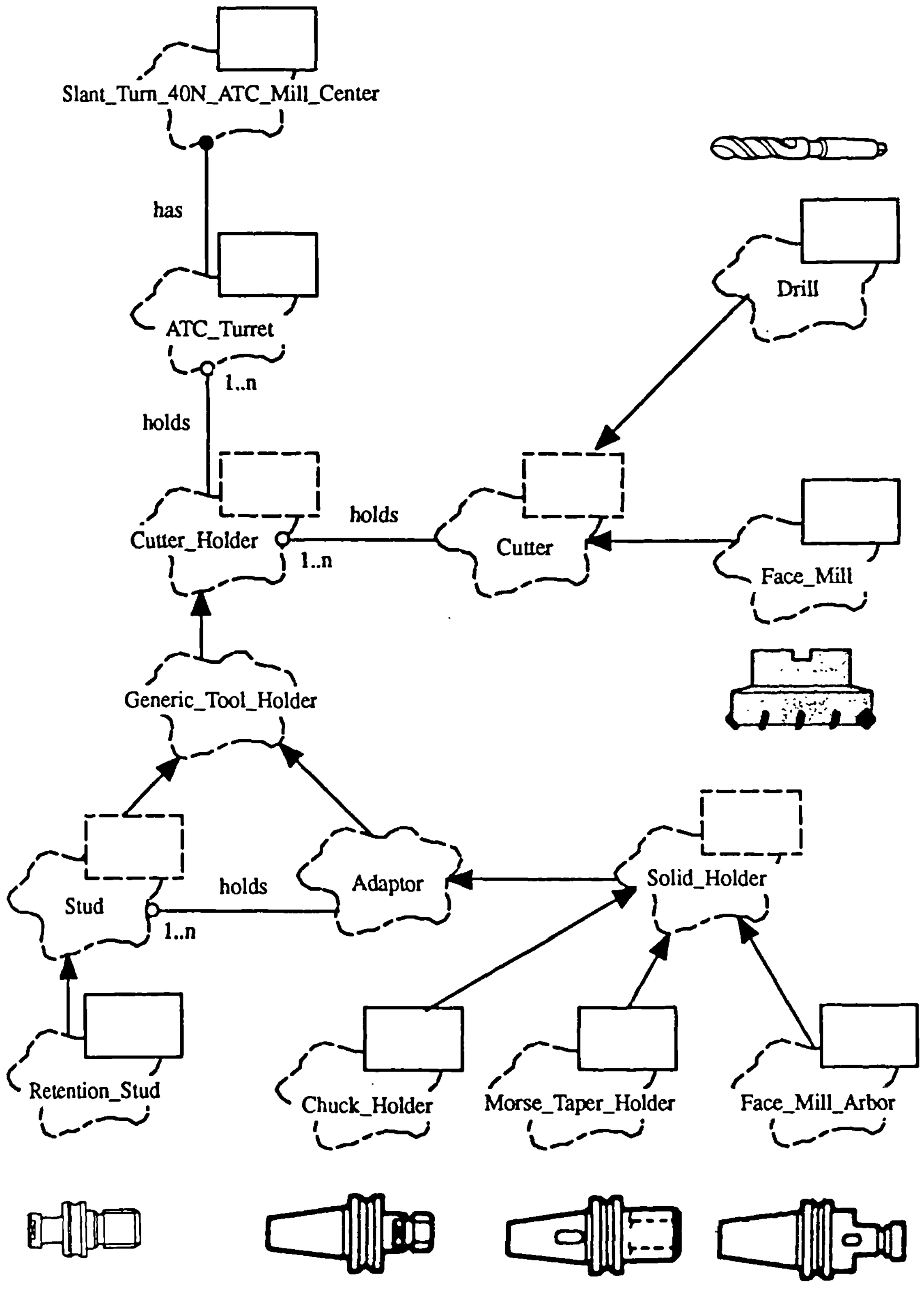




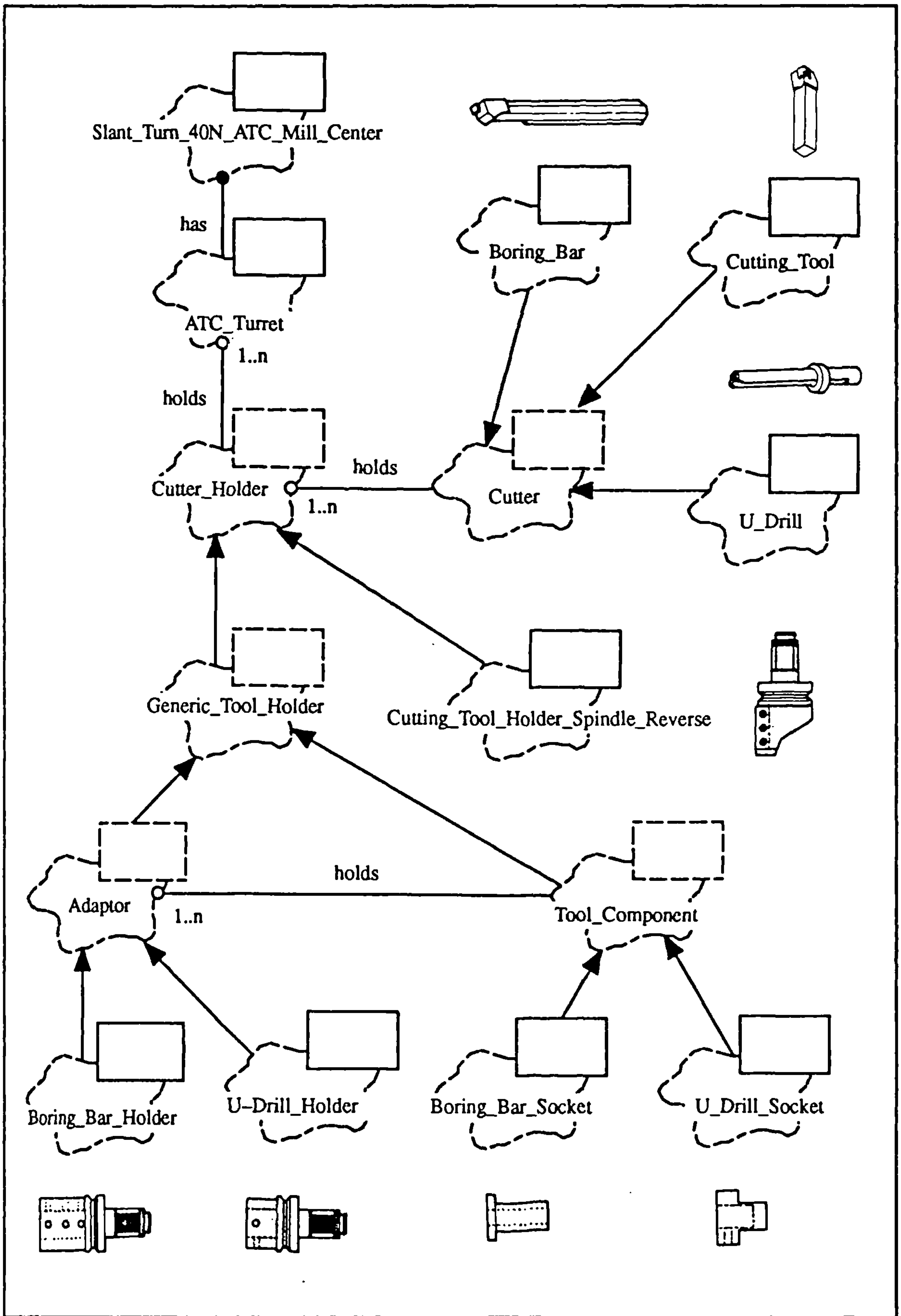
A. Molina | Figure 13.2 Modelling a Turning Centre (Slant Turn 40N ATC Mill Center Catalogue) | LUT–SE Research Group



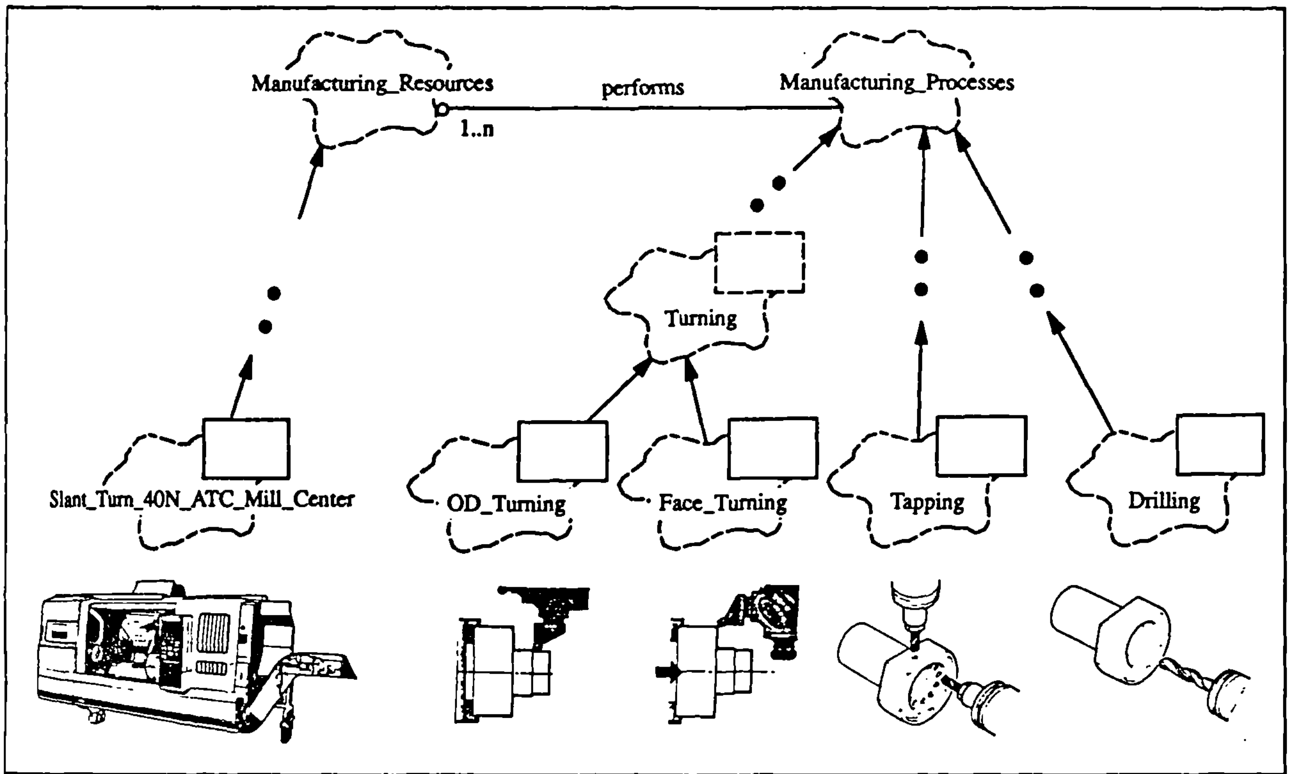
A. Molina | Figure 13.3 Tooling System of the Slant Turn 40N ATC Mill Center (Slant Turn 40N ATC Mill Center Catalogue) | LUT–SE Research Group



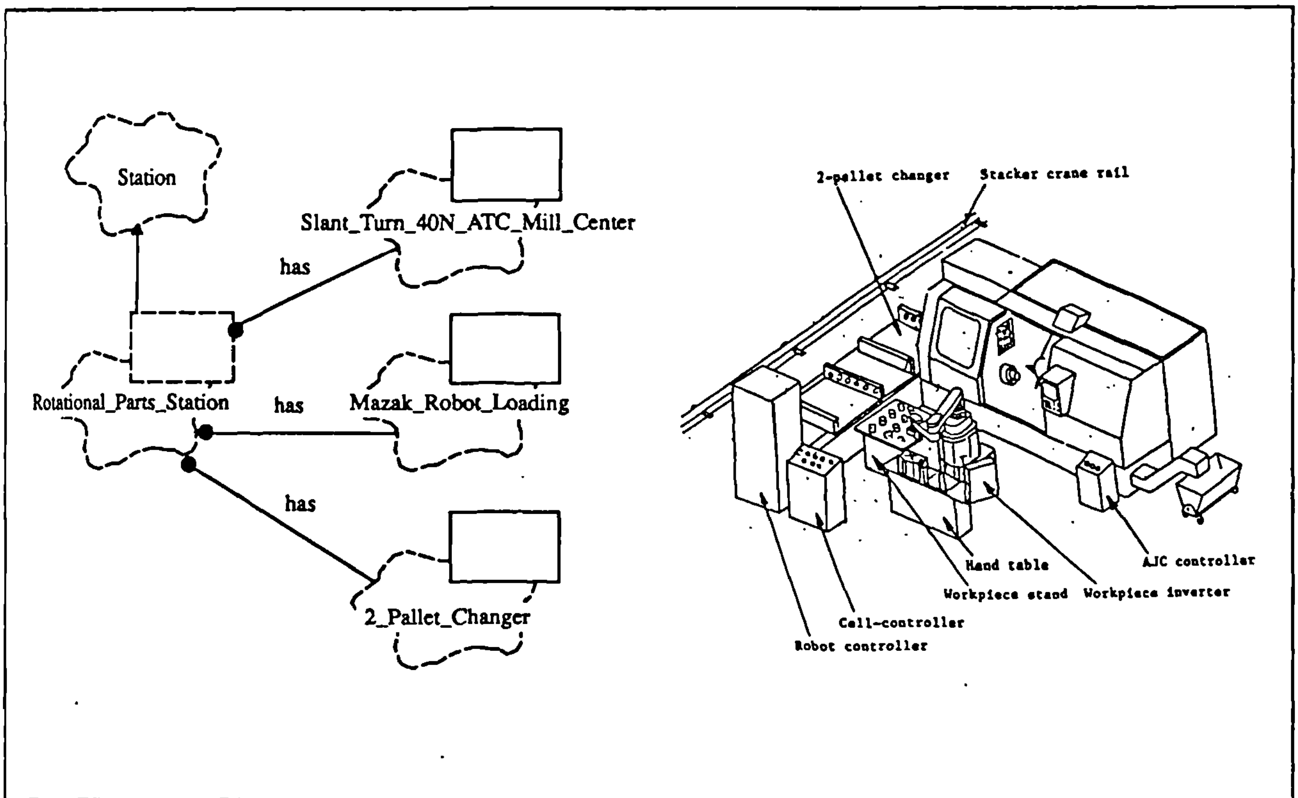
A. Molina	Figure 13.4 Modelling Rotary Tooling System (Slant Turn 40N ATC Mill Center Catalogue)	LUT–SE Research Group



A. Molina	Figure 13.5 Modelling Tooling System (Slant Turn 40N ATC Mill Center Catalogue)	LUT–SE Research Group



A. Molina	Figure 13.6 Modelling the process capability of a Turning Centre (Slant Turn 40N ATC Mill Center Catalogue)	LUT-SE Research Group



A. Molina	Figure 13.7 Modelling the Configuration of the Rotational Parts Station	LUT-SE Research Group

STATION	Facilities Strategies – Focus Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Product Group	OKP, **MP**, FMP, CP	Use Station for Rotational Components which require different machining process If components require only turning process this Station should not be used.	Variety of components (Increase this variety)
Product Volume	HV–HS, HV, LV, **LV–LS**	Operate on batches of minimum 1, maximum n	Cost per part fix regardless batch sizes
Product Life Cycle	**NP**, GP, **MP**, DP	New components require to be rotational with asymetric features Priority given to manufacture components required for the assembly line	Cost per part (if the part is economical to be produced in the Station) Delivery time
Production Mode	P, **F**, **C**, T, L, Co	Use of an informal Group Technology method Assign components to the Station to minimize operations and set–ups.	Number of operations Number of set–ups
Product Group OKP: One of a Kind Products MP: Multiple Products FMP: Few Major Products CP: Commodity Products	Product Volume: LV–LS: Low Volume – Low Standardisation LV: Low Volume HV: High Volume HV–HS: High Volume – High Standardisation	Product Life Cycle: NP: New Products GP: Growing Products MP: Mature Products DP: Declining Products	Production Mode: P: Project (Job Shop) F: Functional (Batch) C: Cellular L: Line Co: Continuos Flow

Table 13.1 Description of the Facilities Strategies of the Rotational Parts Station

STATION	Capacity Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Product Range	HS, S, LS, C	Evaluate potential new components within the Product Range i.e. rotational components with asymmetric features. If new component can be machined at a suitable cost and time then do it. Always increase the number of parts which can be machined in the Station (Now up to 200 different parts).	Cost per part Machining time Components Variety
Cycle Demand	HE, HI, O, S	**If the capability is not enough to meet the deadlines then .... Confidential Strategy**	Delivery Times Meeting of Deadlines
Increase Capacity	O, IS, S, EF, AF	**If the capability is not enough to meet the deadlines then ..... Confidential Strategy**	Due times Cost per part
Product Range HS: Highly Standard S: Standard LS: Low Standard C: Customized	Cycle Demand: HE: Holding Excess Capacity HI: Holding Seasonal Inventories O: Overtime S: Subcontracting	Increase Capacity: O: Overtime IS: Increase Shifts S: Subcontracting EF: Enlarge Facility AF: Additional Facility in a new location.	

Table 13.2 Description of the Capacity Strategies of the Rotational Parts Station

STATION	Technology Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Processes	Turning, Milling, Boring, Chamfering, Tapping, Drilling	Criteria: Minimize Cost, Minimize Lead Time, Maximize Quality Type of Material: cast iron, steel and castings. If component only requires Turning Operations use other Machine	Cost Lead Time Tolerances
Resources	Slant Turn 40N ATC Mill Center Tooling System Mazak Robot 2–Pallet Changer	Maximum Diameter 250 mm – Minimum Diameter 50 mm Minimum Length 15 mm Maximum speed 2500 r.p.m. For large components (> 80 mm) use machine with work cross (steadies) Use jaws which allow a clearance of 7 mm. The tool system should always contain the maximum diversity of tools. Use always standard tooling. Avoid the use of sister tools. Avoid the change of tools, it is better to move the workpiece to another machine. Change tool when the tool life has expired before starting a new job. Each machine has different set of tools but some of the tools are common between machines Use machining parameters as specified by tool suppliers Robot Maximum Weight 45 Kg (4 axis 1 robot) Maximum weight 38 kg (5 axis 2 robots)	Machine utilization Machining times Tool Life
Configuration	**HA**, PA, LA, M PS: **CNC** MHS: **Robot** THS: **ATC** S/WS: **Local Buffer** SI: **DNC–Integration of Multiple FMCs**	Always program in Mazatrol Language using standard templates Avoid the use of the ATC Keep the 2–Pallet Changer fully utilizes	Flexibility ??
Layout	Intracell: **D**	Robot speed and time are not important factors	Flexibility ??

Configuration (Degree of Automation):
HA: Higlhly Automated
PA: Partial Automated
LA: Low Automated
M: Manual

PS: Processing System (Operator–assisted NC machines, FMS, etc)
MHS: Material Handling System
THS: Tool Handling System
QAS: Quality Assurance System
S/WS: Storage/Warehouse System
SI : System Integration (Integration of Multiple FMC, Integration with CAD/CAM, etc.)

Layout (Intracell Type):
L: Line Layout
S: S Layout,
O: O Layout,
D: D (robot served) Layout
W: Layout.
U: U Layout
LL: L Layout

Table 13.3 Description of the Technology Strategies of the Rotational Parts Station

Chapter 14

Case Study: Flexible Batch Manufacture (Part I)

14.1. Introduction

This chapter introduces the facility where the modelling work was undertaken, i.e. the Mazak European Factory; the factory manufacturing configuration and layout is described in some detail in Chapter 15. The information used in the case study has been based on the brochure "Mazak European Factory", articles written by Kurimoto (1988, 1989) and interviews with key people of the Factory. Important issues related to the production control and management of the machining and assembly cells are described.

14.2. The Mazak European Factory at Worcester

The Mazak European Factory at Worcester is a Yamazaki Mazak UK plant dedicated to the production of a variety of machine tools. The output of the factory was originally based on two products, currently a wider range of products are produced. The majority of machines now manufactured are more complex that original products associated with the factory (figure 14.1). To each machine tool could be added particular additional features as requested by the customer.

14.3. Production Planning Lead Times

The factory works in a very flexible mode. The Production Planning for the Flexible Manufacturing System of the factory is based on a one month schedule developed from a four month forward plan. Production Control delivers the information regarding the number and type of machines to be produced and the machine release plan. Using this information a Production Plan is created which contains the information regarding the parts supply and associated lead times. There are five specific manufacturing areas which each are given separated lead times:

- Unit Parts: dates where the sub–assemblies have to be machined in the Rotational Parts Line and Small Prismatic Machining Line.

- Unit Release: dates for the release of the kits which have all the unit parts to be used in assembly to build and test the sub–assemblies, these are kept in the Automated Warehouse.

- A–Line Release: actual dates to machine the bases and columns.

- Cast Bases to Paint: dates when the bases and columns have to go for painting.

- Assembly Line Start: dates when the machine has to go down the line on assembly.

The lead time pull mechanism is represented in figure 14.2 where the due dates of the required production of machine in a month is used as the basis to calculate the Unit Parts, Unit Release, A–Line Release, Bases (Cast and Paint) and Assembly Line Start lead times. All these lead time are different and are according to the time when the different parts are required in the next process. The earliest time is the Unit parts because these parts have to be ready for the subassemblies, which will be used in the assembly line. Once the lead times are calculated work sheets are generated and sent to the different manufacturing areas.

14.4. Production Planning in the Rotational Parts Line

Figure 14.3 illustrates in general how the production planning of the Rotational Parts Line is carried out. A worksheet is generated based on the Production Plan described above using a simulation program. This simulation creates a schedule for the unit parts which are required to be produced to meet the lead times specified in the Production Plan during a month. The results of the simulation are divided into weeks in order to prepare a weekly sub–schedule for the Rotational Parts Line. The preparation of this sub–schedule is undertaken by the Production Controller and the Cell Manager. Subcontracting is considered while creating this sub–schedule. For each job (unit part) a work number is allocated together with information regarding quantity and material requirements.

Once this sub–schedule has been prepared, an order for the material is placed and a detail scheduling program is generated. This scheduling program allocates each job to a machining station. The job requirements are matched against the station capabilities following the following criteria: jaws (sizes and dimensions), required tool set and robot limitations (sizes and dimensions). These decisions are made based on experience. This scheduling program is downloaded to the FMS cell controller and is executed to start the production in the Rotational Parts Line. This program can be overwritten at any time to introduce unscheduled high priority jobs.

With the expanded work on manufacture the original Rotational Parts Line has been given the support of a non automated cell of basic CNC lathes. The discussion is restricted to the Rotational Parts Line and related processes, and does not include the complementary processes which are used in particular components, such as grinding.

14.5. Operation of the Rotational Parts Line

The Rotational Parts Line is a Flexible Machining Systems composed of three Rotational Parts Stations (described in Chapter 13, subsection 13.2.4), an Auto Stacking Crane and a 60 Pallet Stocker. Figure 14.4 shows the configuration of this FMS. This FMS system is controlled by a FMS Mazatrol Controller and the control configuration is presented in figure 14.5. The Rotational Parts Line operates in the following manner:

1. Pallets with raw materials are brought to the Rotational Parts Line from the Warehouse using the AGVs.

2. The pallets are carried into the Rotational Parts Line manually, and are taken by the Auto Stacking Crane to be placed into the Pallet Stocker.

3. Once the pallets have been placed in the Pallet Stocker the FMS Controller schedules the pallets to the Rotational Parts Stations.

4. Before starting the operations in the station the FMS controller checks the following information: pallet is in the cutting buffer, CNC Program for machining the part and Robot program has been downloaded, and finally the schedule for the job has been downloaded to the cell controller (station controller).

5. The robot program starts, this program verifies which hand has been used, if required, change hand (the robot has the capacity of having 3 hands but only 2 are used), pick-up component, and go back to waiting position.

6. The machine changes jaws and opens the door. The robot goes in, places component in the chuck, jaws open, robot places component, jaws clamp the component, robot goes out, doors close.

7. Machine starts CNC program and machining operations, robot pick-ups next component and go back to waiting position.

The raw material is introduced to the Rotational Parts Line to be placed in the Pallet Stocker as it is delivered, it is important to keep in the stocker the majority of raw material required. The Pallet Stocker stores raw material and machined parts. It is also important to machine full pallets instead of incomplete pallets, this decision is taken by the Cell Manager in order to improve the usage of the FMS, and to achieve a faster response in future requests. If an unexpected component has to be produced, the schedule is overwritten, and as soon as the material is available and the station, where the new component has to be machined, is free, the job is sent to be manufactured. In this cases the usually time of response, if the material is available, is the time the station requires to finish its previous job plus the set up and the machining time of the extraordinary job.

14.6. The Assembly Hall

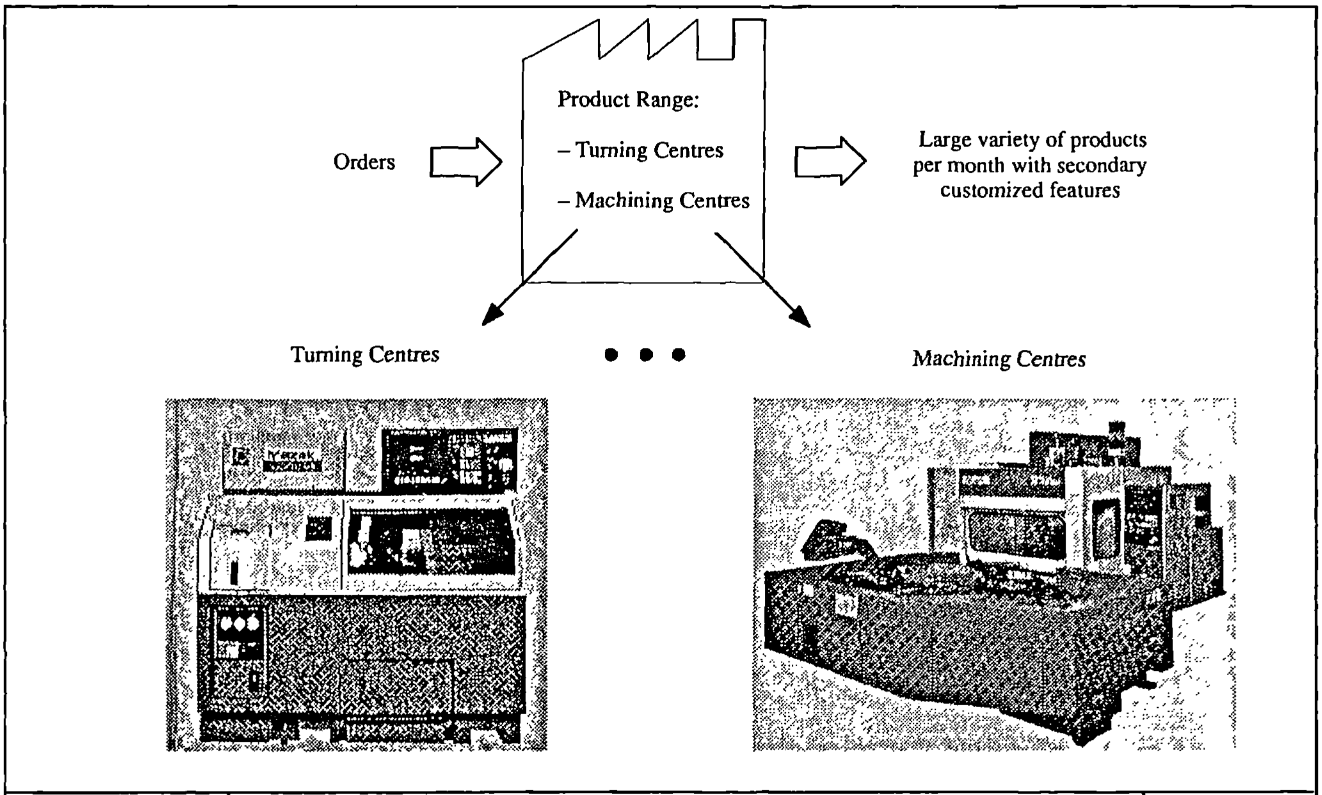
Assembly is the last of all the processes carried out within the factory. The organisation of this line is illustrated in figure 14.6. There are two separated areas, the actual assembly line and the subassembly area. In the subassembly area, called Unit Assembly, all the unit parts which are components of the subassemblies are brought from the warehouse to be assemble. Typical sub–assemblies are the turrets, tool magazines and spindles. Once the subassemblies are assembled and tested, they are send back to the automated warehouse to be part of the machine tool kits for the assembly line.

The assembly line is organised into four different sections: A–Line, General Assembly, Electrical Assembly and Mechanical Adjustment. The inspection and shipping areas are considered to be at the end of the assembly line. For each product group (type of machine tool) a team is assigned. This team is responsible to build up the machine tool and take it through the different sections from the A–Line to the Inspection. Each team is composed of sub–teams which are in charge of the different sections of assembly, i.e. there is a sub–team for A–Line, another for General Assembly and so on. This is a new type of organisation which have produced good results because there are less communication problems between the different sections and each team has a well defined end goal, i.e. the production of a type of machine tool.

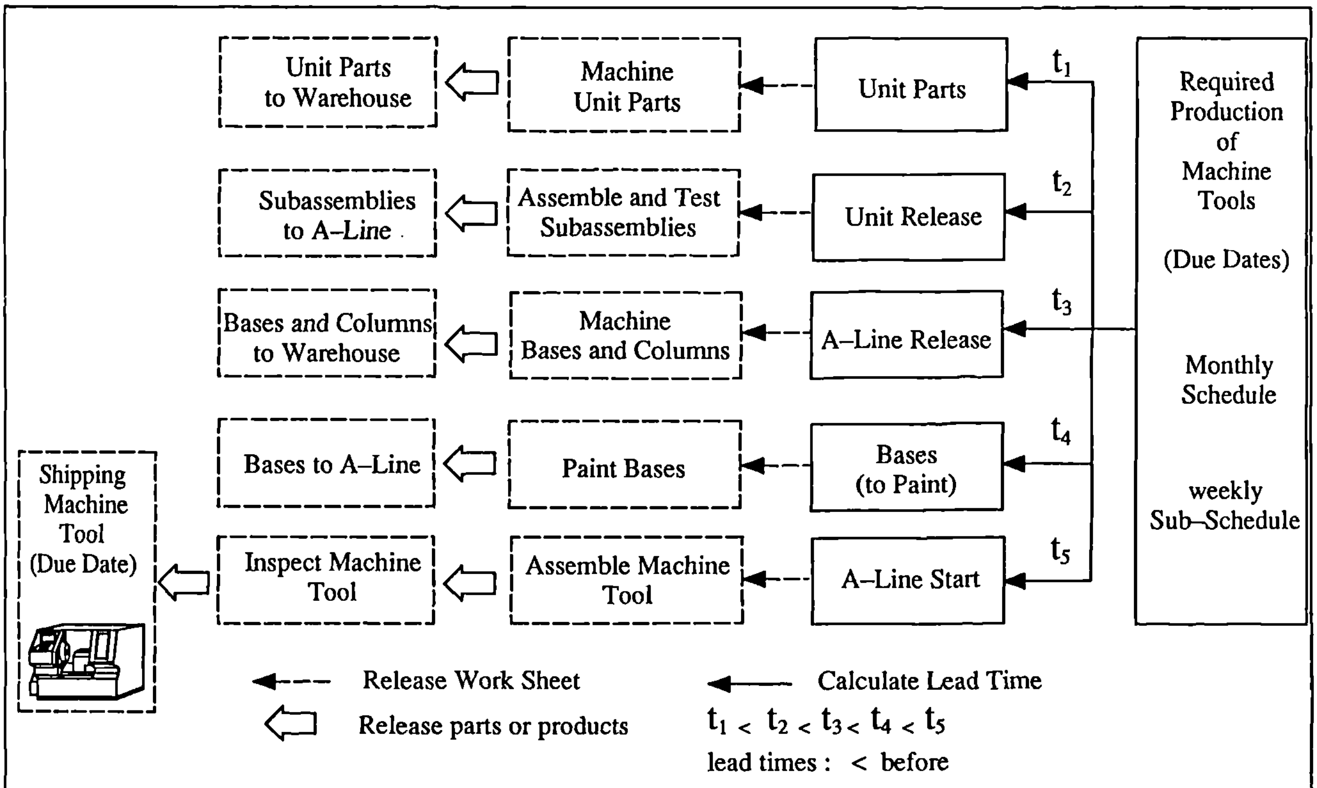
In the A–Line section the alignments of the machine are set, and subassemblies are mounted into the bases and carriages. After A–Line the machine is moved to General Assembly where it remains until is completed and ready to inspection. Instead of moving the machine through the different sections the sub–teams are rotated to the different machines under their responsibility. This is another improvement because it has been proven to be more flexible and more effective to move the different sub–teams around than moving the machine from one section to the other. The general assembly sub–team is in charge of putting the fabrications, hydraulics, pneumatics to the machine and at the same time to build up the the shell of the machine. All the electrical components are set up by the electrical sub–team and finally the mechanical adjustments are carried out by the last sub–team. From this point the machine is taken to inspection which normally takes 48 hours, 24 hours of running test and 24 hours for final adjustments. The machine is then cleaned and packed to be sent to the customer.

Even when assembly is an easy process to understand, its control is quite complicated. The reason of this is the number of inputs the assembly process receives from other areas and from external suppliers. Figure 14.7 shows a generic representation of the assembly process where the inputs identified are machined parts (from the Machining Shop), subassemblies (from the Unit Assembly), sheet metal parts (from the Sheet Metal Working Shop), painted parts – bases (from the Painting Shop), and procured parts (from external suppliers). The control of the process is based on two documents, the machine specification which is a customer order with the detailed configuration of the machine, and the work sheet which is the production plan with the due date of each machine. The resources used by the assembly process are the assembly teams and the tools required to perform the assembly tasks, some of these tools are specialized.

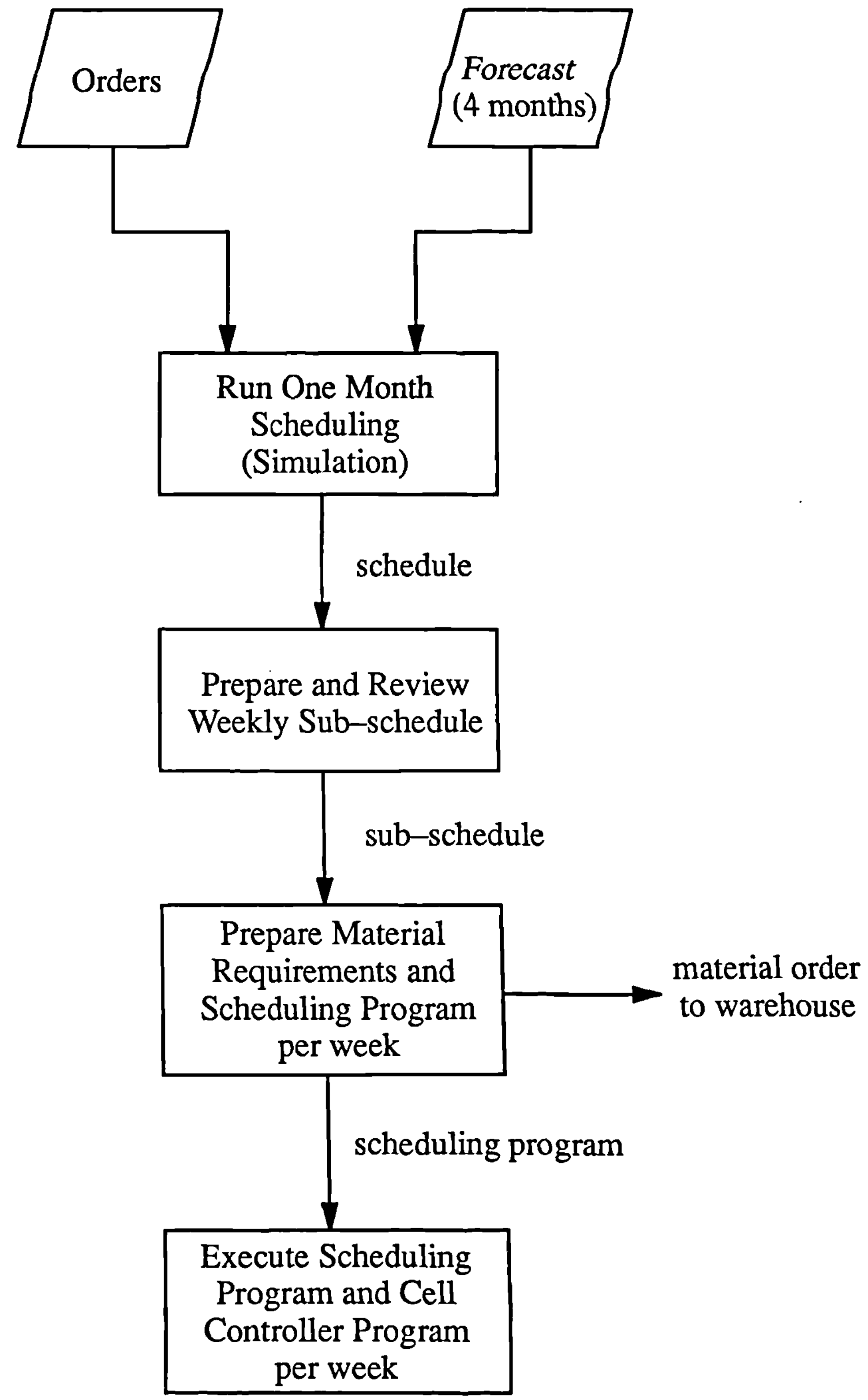
Because assembly is a customer area of the factory, this area could be in big trouble if the in–house and procurement parts do not arrive on time. Therefore a strictly control over suppliers is required. The new range of machines produced requires quite a lot additional time to be assembled, an example given by the Assembly Manager was that to build 4 or 5 simple turning centres will take the same time than to assemble one of the more complicated turning centre configurations. The organisation of the assembly line in teams assigned to specific machine types has allowed to have better indicators of how the assembly line is performing. As the team leaders can keep a record of how long it takes to assemble the different type of machines then they can evaluate how well the actual production of a type of machine is being performed. By using these performance measures the team leader can move people to the sections with bottle-necks. New people are assigned to the different sections of the assembly line according to their background experience, in this way the learning curve is reduced.



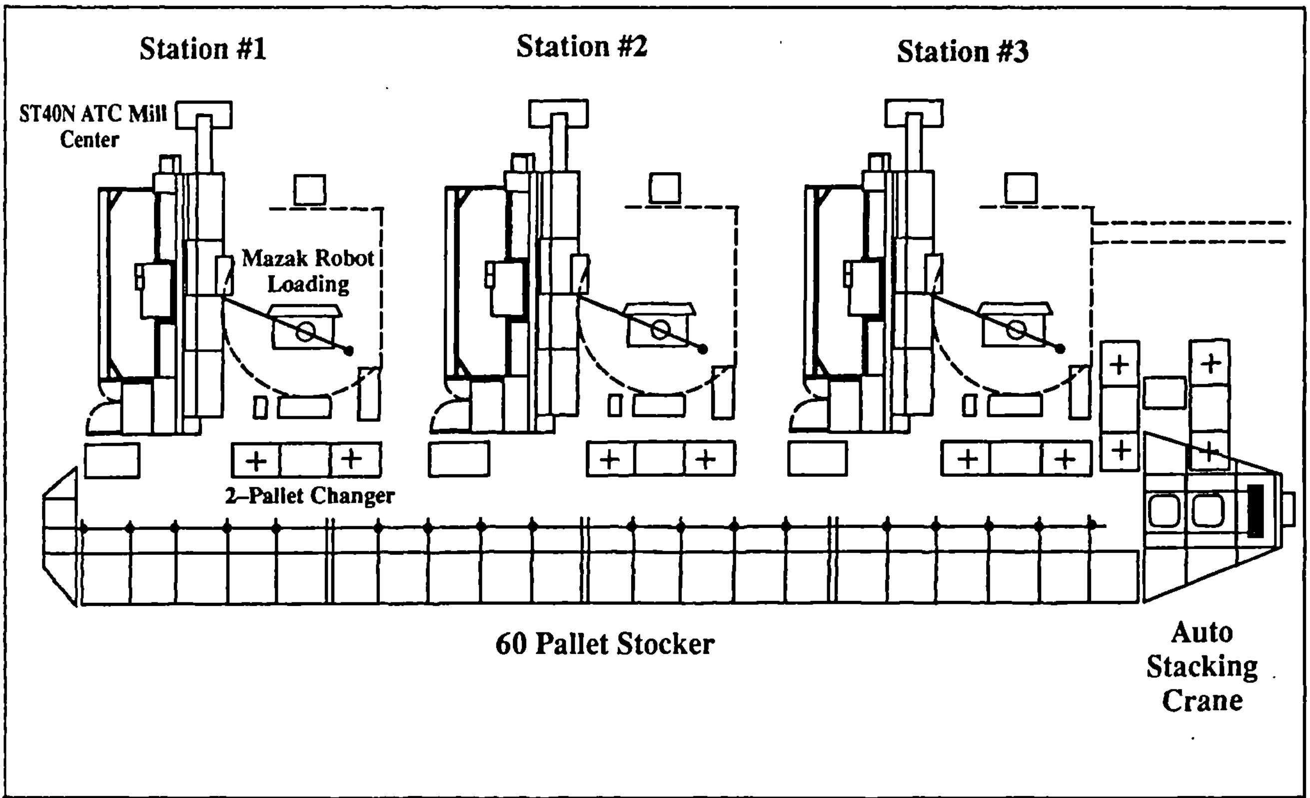
A. Molina | Figure 14.1 The Mazak European Factory at Worcester | LUT–SE Research Group



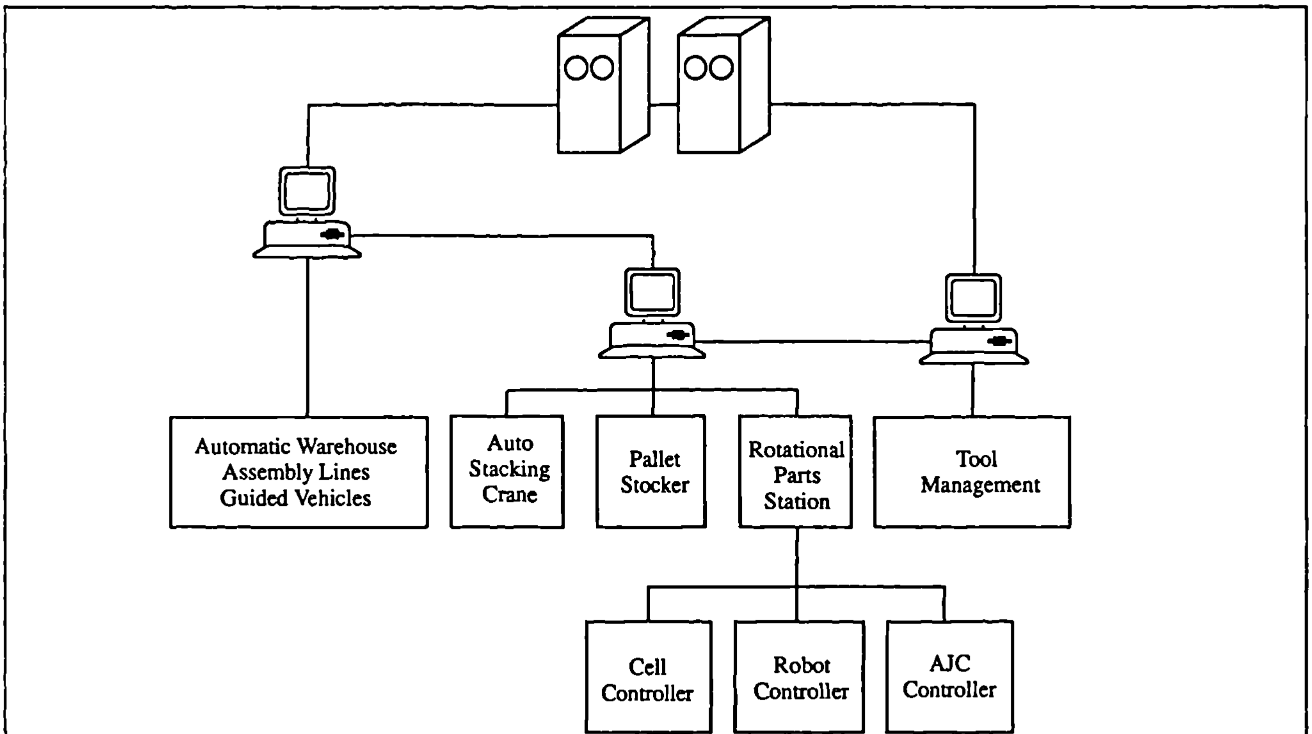
A. Molina | Figure 14.2 Pull Mechanism for Production Lead Times | LUT–SE Research Group



A. Molina	Figure 14.3 Production Planning of the Rotational Parts Line	LUT–SE Research Group

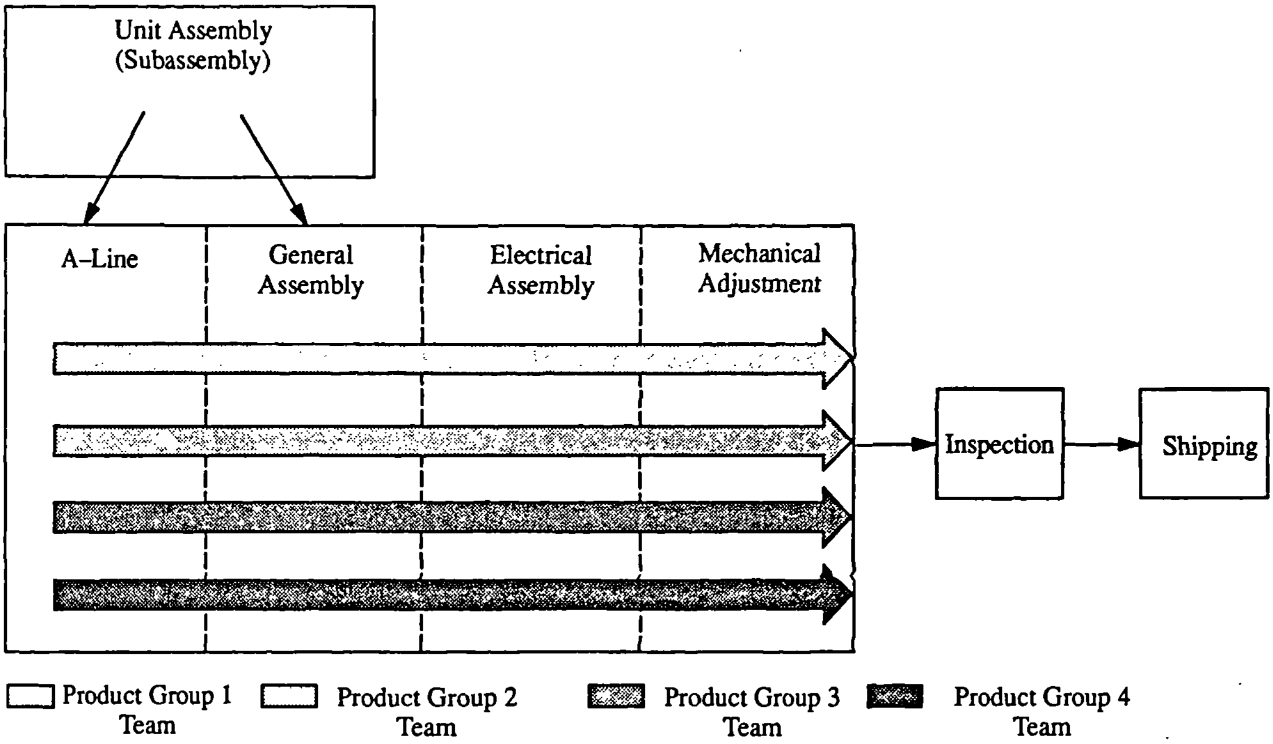


A. Molina | Figure 14.4 Rotational Parts Line Configuration | LUT-SE Research Group

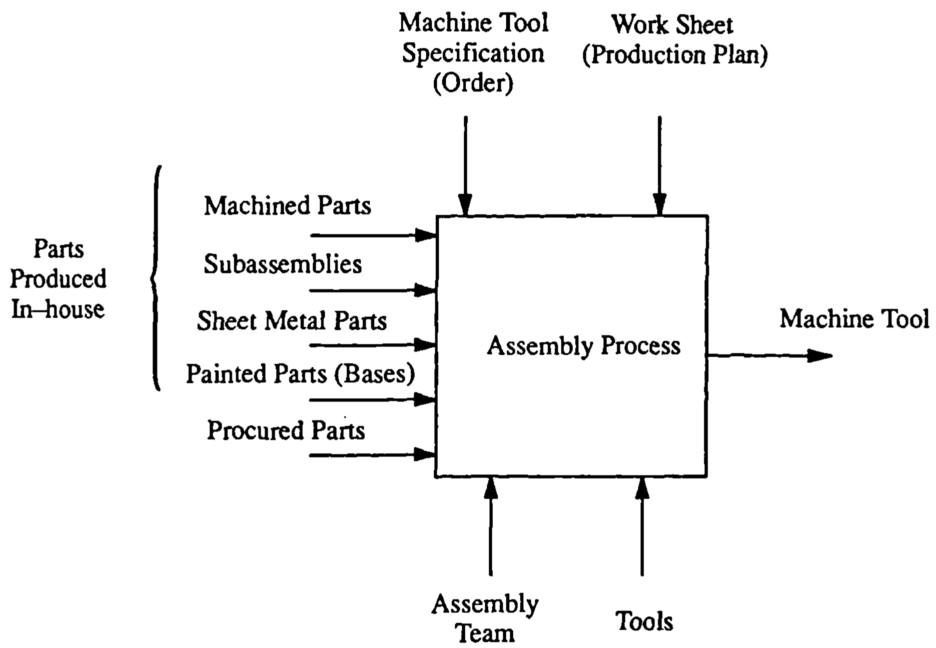


NOTE: AJC – Automated Jaw Changer

A. Molina | Figure 14.5 Computer Control Architecture of Worcester Factory and Rotational Parts Line (Adapted from the MAZAK Factory Video) | LUT-SE Research Group



A. Molina | Figure 14.6 The Organisation of the Assembly Line | LUT-SE Research Group



A. Molina | Figure 14.7 The Assembly Process Generic Definition | LUT-SE Research Group

Chapter 15

Case Study: Flexible Batch Manufacture (Part II)

15.1. Introduction

This chapter describes an adequate detail for the need of this thesis the layout of the Mazak Factory. The cell model of the Rotational Parts Line is presented with its associated strategic decisions and operational rules. The research work related to the exploration of how to model the assembly process in the Manufacturing Model is reported as well.

15.2. The Mazak European Factory Layout

The factory is composed of different functional areas and has been divided into 15 areas, see figure 15.1. This research has focused in modelling the Rotational Parts Line (3) and the Assembly Hall (11). The raw material required by the Rotational Parts Line is released by the warehouse and is carried to the Rotational Parts Line using AGVs. After the units parts (sub–assemblies) have been machined, they are sent to the Automated Warehouse (8) to build the kits required to assemble the machines in the assembly line. The kits are released to the assembly area via AGVs when required.

15.3. Partial Representation of the Factory using the modelling methodology

The Integrated Manufacturing Information Methodology (described in Chapter 11 and 12) was used by the author to build a consistent model of the functional areas of the Mazak European Factory of major interest in this research i.e. the Machining and Assembly Shops. To reduce the complexity of the work, a partial model of the factory was built and divided into 8 shops: Machining, Superfinishing and Quality Control, Sheet Metal Working, Assembly, Precision Assembly, Painting, Automated Warehouse and Transporter AGV (Figure 15.2).

The Machining Section Layout showed in figure 15.3 constitutes the Machining Shop of the partial factory model and has been decomposed further into cells and stations. The Machining Line in the model is organized in three cells: Large Prismatic Machining Line, Small Prismatic Machining Line and Rotational Parts Line (figure 15.4). Furthermore, each of these cells is decomposed into stations. The Large Prismatic Machining Line *includes* three Large Prismatic Machining Stations, the Small Prismatic Machining Line is composed of seven stations, and finally the Rotational Parts Line *includes* three Rotational Parts Stations.

15.4. Modelling the Rotational Parts Line

For the purposes of modelling the Cell Level of the Manufacturing Model the following features have been highlighted: the resources in the Rotational Parts Line, the processes carried out within the cell and the strategies which govern the operation of the cell. The model of the Rotational Parts Station presented in Chapter 13 Section 13.2.4 with the configuration of a ST 40N ATC Mill Center, Yamazaki Robot Loading and a 2–Pallet Buffer has been re–used in the model of the cell and is included as a station. In addition to the stations the following elements have been modelled: 60 Pallet Stocker and Auto Stacking Crane. In Chapter 14 (figure 14.4) the configuration of the Rotational Parts Line with all these elements has been presented. Therefore the model of the Rotational Parts Line *includes* three Rotational Parts Stations (with their respective resources) and *has* the following resources Auto Stacking Crane and Pallet Stocker, see figure 15.5.

The processes within the Rotational Parts Line which have been modelled are: the machining process which can be performed by the Rotational Parts Stations (e.g. turning, milling, drilling, etc.) and the process executed by the Auto Stacking Crane, i.e. Loading, Transferring and Unloading (Figure 15.6).

15.5. Modelling the Strategies of the Rotational Parts Line

A very important aspect of the Rotational Parts Line is represented by the strategies. As explained in Chapter 12, Section 12.6, the strategies must represent how the manufacturing facility is structured and used to support the realization of the manufacturing function in order to achieve the manufacturing objectives of a company. Therefore the strategies related to the Rotational Parts Line should describe these aspects of the facility. This is very important because the way a facility is structured, organised and operated will enable the company to pursue its chosen competitive strategy over the long term. This ability to model strategies is one of the novel elements of this research. The strategies associated to the Rotation Parts Line belong to the following four categories: facilities, capacities, technology and production planning/material control.

The cell (Rotational Parts Line) is focused to machine multiple components which will be used in the subassemblies of the machine tool. Because the strategies between the Stations and Cell Levels of the Manufacturing Model should be consistent. The Rotational Parts Line should be used only in components which require in addition to turning other machining process such as: milling, tapping, etc. If components only need turning then the stand alone turning centres should be used. The cell performance is measured by analysing the variety and number of components which can be machined at a given time, and the component complexity. The Product Group has been therefore defined to be Multiple Products, which conforms with the station strategic decision. The batch sizes can be of a minimum of one job (but full pallets) to n jobs consequently the Product Volume was described to be Low Volume –Low Standardisation. In a similar manner that the station the cost per par should be kept fix regardless the batch size, this is the most important performance measure. Another performance measure is the different number of components which can be machined at a given time period. The cell is mainly machining new components and maturity components (unit parts – sub–assemblies). The principal characteristic of the new components is that they should be rotational with asymmetrical features, and their machining cost should be competitive against subcontractors. On the other hand, the maturity component are the sub–assemblies required for the actual production of the different types of machines produced within the factory. Unexpected orders will have a high priority, therefore they should be schedule into the cell as soon as the material is available and the station required is free. Some performance measure are number of new components, lead times and cost of new components. Finally the Production Mode has been described to be Batch and Cell. Even when there is no standard group technology method, each station in the Rotational Parts Line has different tool and jaws configuration, therefore the range of products each of the station can machine varies. The complete set of facility strategies drawn from the study of the Mazak Factory is described in Table 15.1.

Turning now to capacity strategic decisions, these have been classified as Product Range, Cycle Demand and Increase Capacity. These strategies represent how the capacity of the cell is managed. The Product Range is variable, up to 200 different parts are machined in this cell, therefore the range has been defined as Low Standard, i.e. the product range is wide. Parts can be machined to meet the customer requirements, in case there is a need for an unexpected part, this can be manufactured if the part belongs to the product range. The judgment of this possibility is made by the cell manager and staff. The cycle of demand is meet by working up to the peak capacity of the machining cells. The performance measures for these strategies are: volume, number of components in a given time, capacity utilization and space utilization. Table 15.2 presents the above capacity strategies.

The technology strategies manages and controls the use of the manufacturing processes and resources. These strategies are grouped in four categories: Processes, Resources, Configuration and Layout (Table 15.3). The following technology strategies associated to processes: type of material to be used (e.g. cast iron, steel and castings) and the constraint of avoiding the use of the Slant Turn 40N ATC Mill Center if a component requires only to be turned. Restrictions or rules can be applied to the use of the cell, for example, the definitions of maximum and minimum diameters, minimum length, and maximum weight. Some these restrictions are closely related to the ones defined for the Rotational Parts Station in Chapter 13, Subsection 13.2.5. The configuration variable specifies the important decisions made related to the configuration such as the definition of the processing system – FMS with three stations, the material handling system – Auto

Stacking Crane, tool handling system – Automatic Tool Changer, storage and warehouse system – Pallet Stocker and system integration with DNC and multiple FMCs. Some rules associated with these decisions are described in Table 15.3. Finally, the variable related to Layout has been defined as Line Layout. In this type of layout the speed of the Auto Stacking Crane is not important for the performance of the cell, and the function of this resource is to serve Stations and Pallet Stocker. The performance measure chosen for the configuration and layout decision is flexibility.

Finally the production planning and material control strategies define how the cell is operated. These type of strategies were not defined for the station level because in the model of the station level (Chapter 13, Subsection 13.2.5) there was no need to define these strategies. The first variable is related to the Master Scheduling. The cell works in two different manners: Make to Order and Engineered to Order. Nowadays the production of the factory is make to order, therefore all the components produced by the Rotational Parts Line are already to be used in a machine tool. As the factory has the capacity to deal with customized configuration, the factory has the characteristic to work in the mode of engineered to order. The shop flow control is of pull type where everything is pulled by the due dates of the machines to be released, and therefore the lead times of the assembly line influences the remainder lead times. The Material Planning is time–phased as all the materials required are ordered on weekly basis. Finally the Cell Scheduling uses a type of group technology allocation, where the components are matched against the capabilities of each station, where the important factors are the jaws, tools and robot characteristics. From these factors, the tool set and robots limitations made the jobs to be allocated in the machine which can deal with the work, in the case of jaws because for each cell three carousels of 15 jaws are kept, the jaws change is possible and therefore the job can be allocated to any cell. Another important rule is the allocation of large components to the station which has a Slant Turn 40N ATC Mill Center with work cross (steadies). These strategies are summarised in Table 15.4.

This model shows how a good characterisation of a manufacturing facility is possible using the Manufacturing Model. Technical information is captured by the generic classes of resources and processes, and company specific information is described using the strategies. The strategies therefore represent how the company is using its manufacturing capability to pursue its chosen manufacturing strategy. The strategies element of the Manufacturing Model will be used in the future for the formulation of the manufacturing strategies as they represent the current manufacturing situation in terms of strategic decision variables. This work will be further explored by the research on "Linking Business Strategies with Manufacturing Performance".

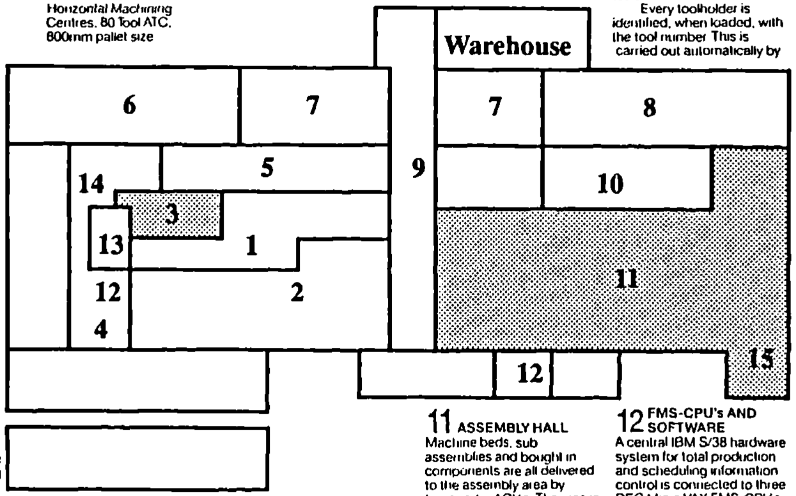
15.6. The applicability of the Cell Level of the Manufacturing Model

The Cell Level of the Manufacturing Model has the potential to be used for the production planning and control of the Rotational Parts Line. As it stands now, the author believes that the Cell Level is appropriate to assist the Cell Manager in the creation of the job list for the scheduling program. However there is a need to developed further three important aspects of the model. The first one is related to the representation of the strategic decisions, operational rules and performance measures in the database in a way that could be used by automated applications to make decisions. These instances in the Manufacturing Model prototype are only implemented as text strings, therefore a more dynamic implementation structure is needed. Second, the behaviour element of the Manufacturing Model needs to be explored in more detail, this aspect will allow real time issues to be considered. Finally, in order to fully support real time control the Manufacturing Model requires to have access to real time data to be able to keep the status, state and conditions of the situation of a facility through time.

15.7. Modelling the Assembly Hall

This activity is of necessity a very limited study, which is focused essentially to gain insight into the future possibilities in the topic. It has proven to be a useful first activity which allows encouraging conclusions to be generated. The model is therefore a partial one and only basic instances have been created to demonstrate how the Manufacturing Model could model the assembly processes.

The Assembly Shop of the partial factory model has been decomposed into four cells: Subassembly, Assembly Line, Inspection and Shipping (figure 15.7). The Assembly Line is then organised in four different Assembly Line Sections, i.e. the Assembly Line *includes* Assembly Line Sections. The four Assembly Line sections are: A–Line, General Assembly, Electrical Assembly, and Mechanical Adjustment; as described in Chapter 14, section 14.6 (figure 14.6). In order to represent these sections the Assembly Line Section has been modelled as a generic class (figure 15.8) with the following attributes. The line *has* Manufacturing Resources which are the team of workers and the assembly tools. It also *has* Manufacturing Processes which in this case will be the different assembly processes carried out by the teams and the transportation process; i.e. the movement of the machine tool from the A–line to the General Assembly Section. The generic process definition described in Chapter 12, Section 12.5, enables the representation of different type of processes carried out in the assembly line. These representation is based on defining the following inputs/controls/outputs of the process: resources, information and material. It is important to mention that no strategies were captured for the assembly sections, but in order to have a consistent model the strategies class is also associated to the Assembly Line Section class, see dashed line in figure 15.8. The instances of the Rotational Parts Line and the partial assembly model can be found in Appendix I.



1 LARGE PRISMATIC MACHINING LINE

Large prismatic parts such as machine beds are machined on three travelling column machining centres each with an 80 tool magazine 36 pallets are stored holding fixtures and workpieces and transfer is carried out by automatic rail guided carts Pallets are identified individually at the time of fixturing by the scheduling computer Each pallet has a built in programmable micro-chip which stores the appropriate identity

Equipment
- 3 Mazak YMS 400 travelling column Machining Centres. 80 tool ATC, 1600 x 3000mm pallet size
- 36 Pallet Stocker.
- Rail Guided Cart.
- 1 Sumitomo CNC Bedway Grinder.

2 SMALL PRISMATIC MACHINING LINE

Machined on seven machining centres components for the gearboxes etc are fed from a 3 tier stacker on pallets together with fixtures and workpieces Transfer to and from the machine tools is carried out by two automatic stacker cranes 30% of components will be fixtured automatically with a newly developed robotic loading system

Equipment
- 7 Mazak H25N *Horizontal Machining* Centres. 80 Tool ATC. 800mm pallet size
- 150 Pallet Stocker
- 70 Buffer Pallet Stocker
- 2 Auto Stacking Cranes
- Washing Station
- DEA 3D Co ordinate measuring machine
- Automatic component location unit

3 ROTATIONAL PARTS LINE

3 Mill Centre Lathes are used to machine rotational parts The machines are fed by Mazak robots from pallets of stacked components

Both normal turning and auxiliary machining such as pitch centre holes slots keyways and cross holes can be machined in a single chucking

Automatic jaw changing provides versatility to machine a wide variety of workpieces

Equipment
- 3 Mazak SI 40N ATC Millcentre lathes 80 Tool ATC Mazak Robot Loading
- Auto Jaw changing (15 Jaws)
- 60 Pallet Stocker
- Auto Stacking Crane

4 TOOL PRESET AND AUTOMATIC TOOL DISTRIBUTION HIGHWAY

Preset tools required are identified on CPU for selection and delivery on the *tool distribution* highway The highway carries tools on an overhead monorail using random access order to replenish tools directly into all the machining centres and lathes and to return worn tools.

Every toolholder is identified, when loaded, with the tool number This is carried out automatically by the system computer coding a programmable micro-chip built into the pull stud

5 SUPERFINISHING AND QUALITY CONTROL – TEMPERATURE CONTROLLED

Using high accuracy CNC *grinding and jig boring* machines roundness and precision can be controlled to levels of less than 0 5 μm Temperature control of ± 1°C ensures constancy during these high precision operations Measurement of roundness, roughness and 3-dimensional levels can be made using ultra-accurate measuring equipment Quality information is fed back to the central CPU for logging

Equipment
- Taly Rond Roundness Measuring
- Taly Surf Surface Measuring
- Zeiss 3D Co-ordinate Measuring.
- 2 Mazak Jig Centre ATC
- Voumard CNC Internal Grinding
- Studer CNC External Grinding
- Jones & Shipman Surface Grinding

6 SHEET METAL WORKING

Fully computer-controlled, sheet metalworking can be performed in approximately half the time taken by conventional factory methods using the latest in laser and CNC machining methods Two Mazak Laser paths are used for sheet metal cutting

Equipment
- 2 Mazak Laser Path Cutting Machines
- CNC folding and bending machines.
- Automatic Welding Machines.
- Storage for 180 part varieties.

7 PAINTING

Two painting systems are provided, one for "cast iron" components and one for sheet metalwork Both systems use automated transport of parts through the spraying areas and the drying stages. Actual spraying will be done manually

8 AUTOMATED WAREHOUSE

The parts loading centre controls the passage of work from the machining to the assembly departments. Under CPU control machined parts, purchased goods and assembled units are held in an automatic warehouse and distributed to the assembly areas using two auto guided transporter trucks 4420 pieces weighing up to 1,000 kg can be stored at any one time

9 MAIN AISLE AND TRANSPORTER AGV's

Used for transporting raw materials and machined goods on two AGV s from the material centre to the machining lines, paint areas and assembly area the main aisle separates the machining and assembly areas of the factory

10 CLEAN ROOM PRECISION SUB-ASSEMBLY

Controlled at a temperature of 19 degrees ± 1 C the air is cleaned to class 10,000 i.e. as clean as the air in the middle of the Pacific Ocean. Wearing special overalls operators assemble spindles and headstocks for NC lathes, and spindle cartridges for machining centres

11 ASSEMBLY HALL

Machine beds, sub assemblies and bought in components are all delivered to the assembly area by transporter AGV s The area is kept to a standard temperature ±2½°C as is the entire factory, while the machines are built and tested

Machines can then be directly despatched or *moved to a prove out area* when fixturing and tooling is ordered by the user

12 FMS-CPU's AND SOFTWARE

A central IBM S/38 hardware system for total production and scheduling information control is connected to three DEC Micro VAX FMS CPU s which control the on-line systems All the lines are *synthetically* controlled by the latest production management software from the moment the raw materials enter the factory to *the despatch of the finished* product One Microvax schedules the automatic warehouse and assembly requirements A second *Microvax schedules* the FMS lines and the third organises the entire tool management including tool offsets, life and replacement

Complete production scheduling, FMS scheduling, tool monitoring and overall plant monitoring software has been developed by Mazak Engineers from many years of experience in building and operating flexible manufacturing systems.

13 UNDERGROUND COOLANT AND CHIP SYSTEM

The automatic *underfloor* chip and coolant *system* carries 30 tonnes of coolant and is able to handle up to 20 tonnes of swarf per day The underfloor coolant and chip disposal system carries the used coolant to a collection tank where the swarf is separated and automatically transported outside the building for collection

14 CENTRAL LUBRICATION

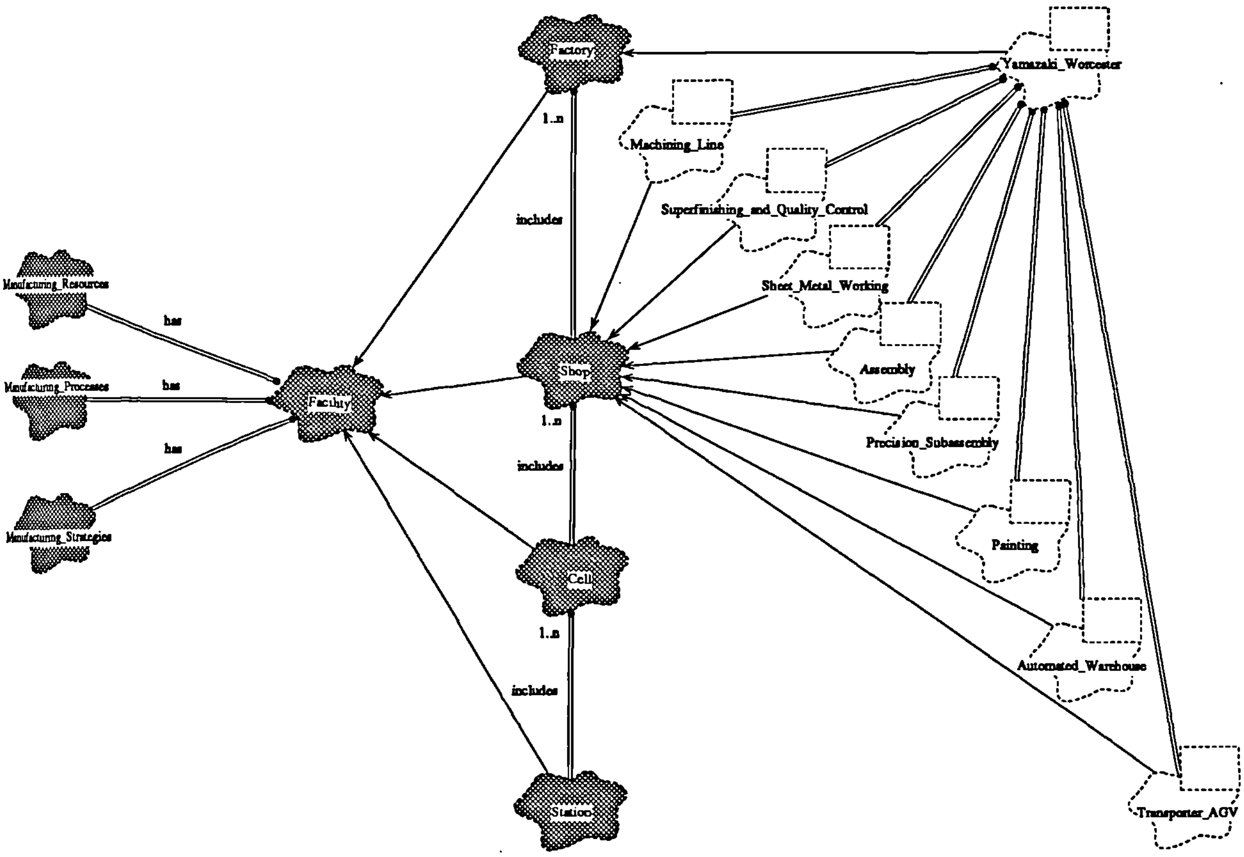
Supplying from a central lubricant tank to all the machines on the production lines, the integrated lubrication system not only eliminates oil shortage and machine stops but also makes oil control easier, saving labour time

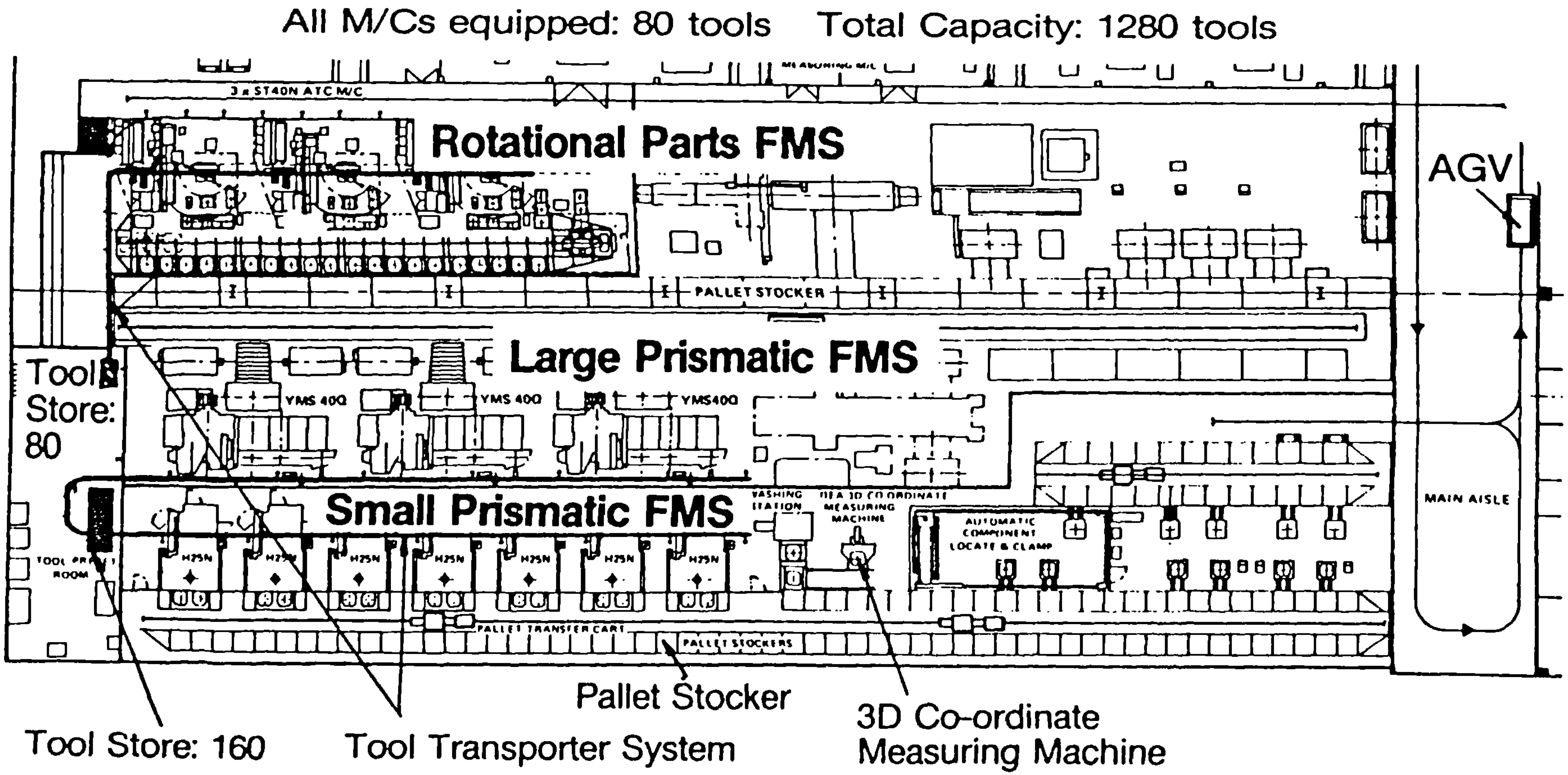
15 DESPATCH

An airlock entry system on despatch ensures temperature control within the factory, while transport is entering and exiting from the main factory

A. Molina | Figure 15.1 The Yamazaki Mazak UK Plant at Worcester (Adapted from brochure "Mazak–Plan of Action") | LUT–SE Research Group

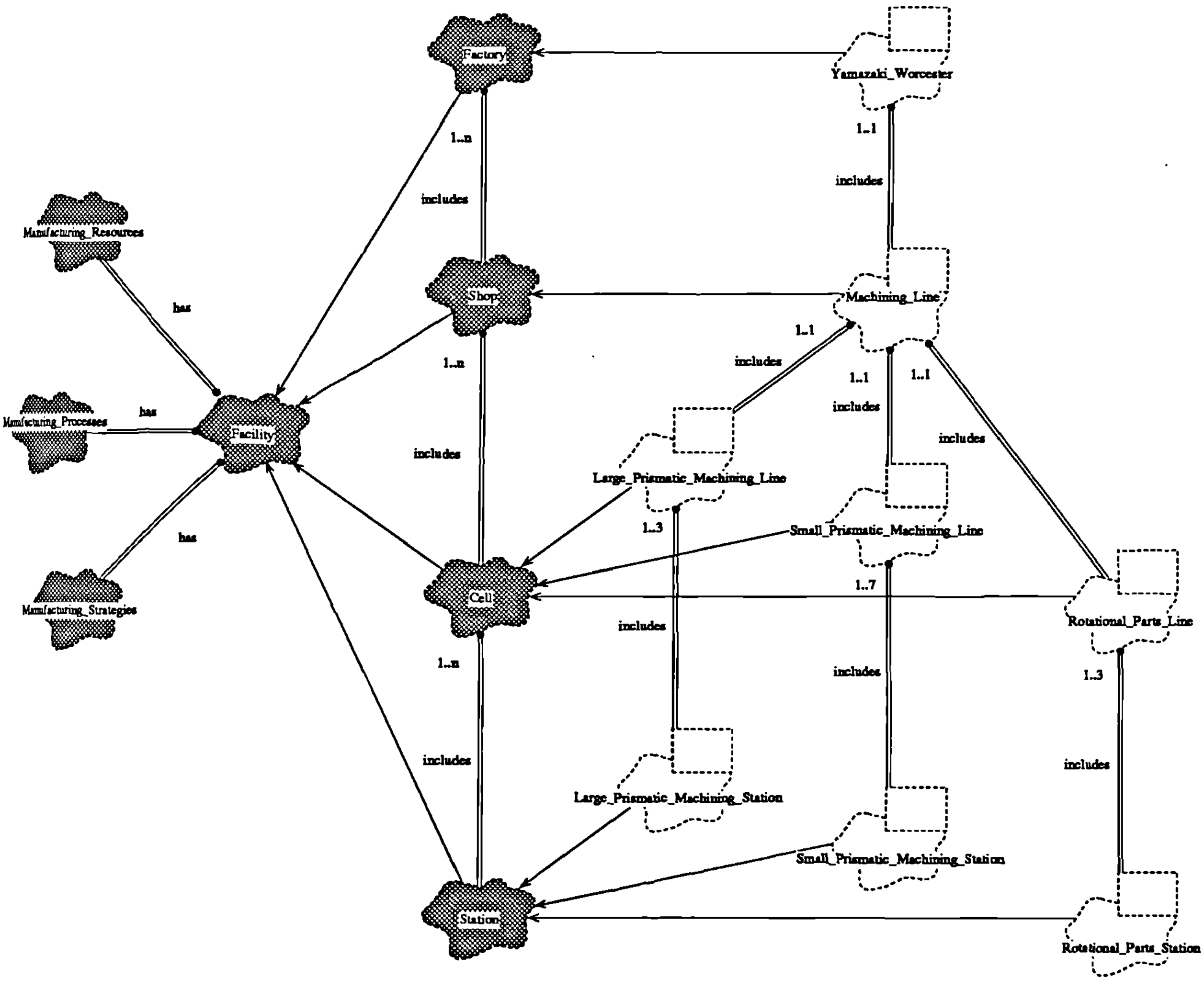
Project Team: LUT-SE Research Group
Title: Figure 15.2 Mazak Factory Partial Model
Project: Manufacturing Model
Printed by: A. Molina

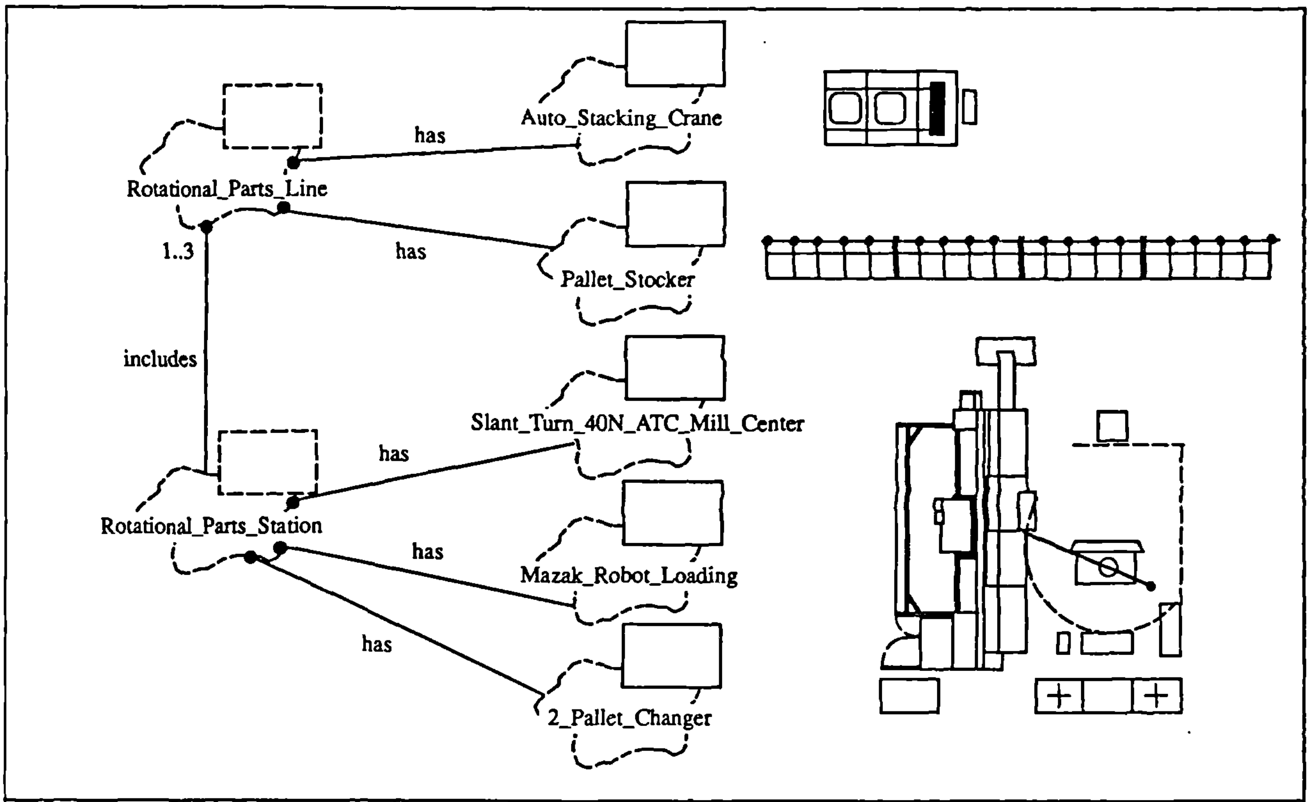




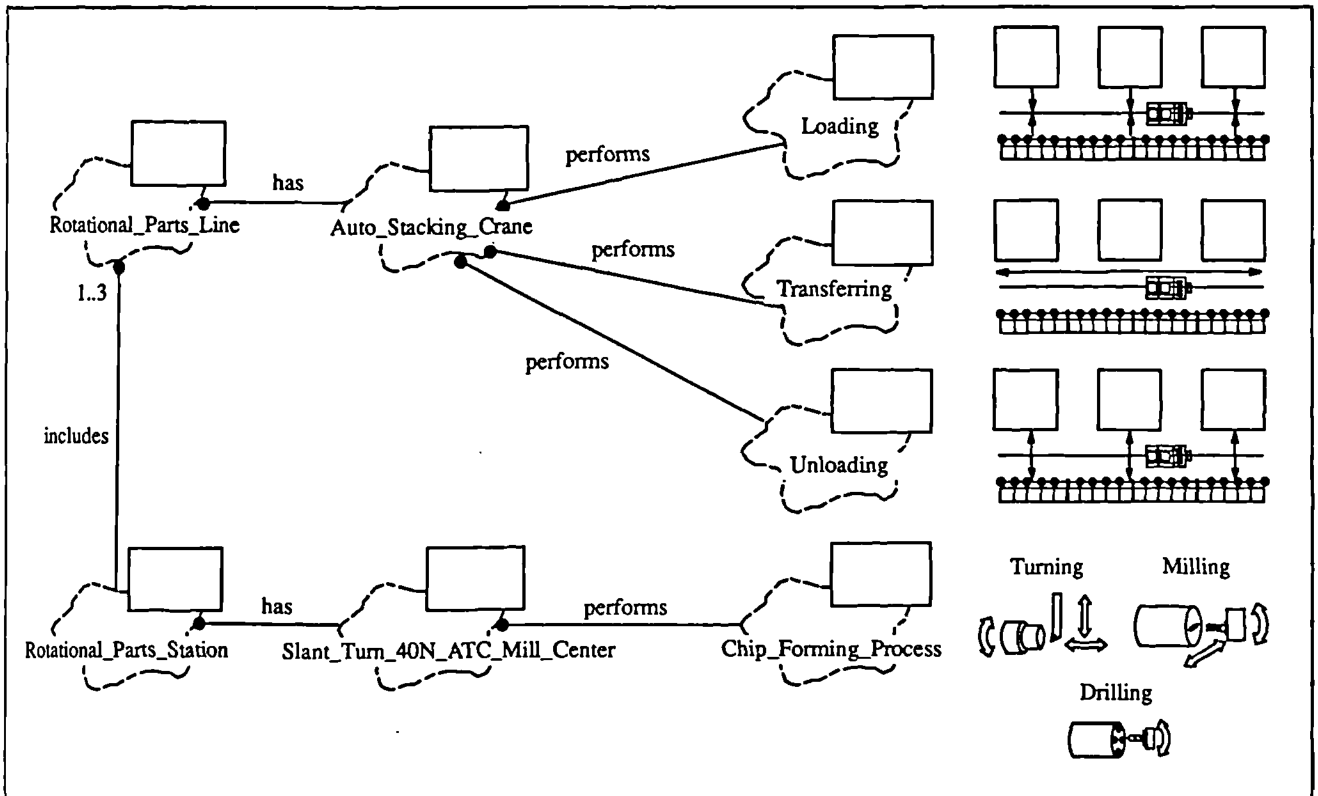
A. Molina	Figure 15.3 The Machining Section Layout (source: Kurimoto A., 1989, "Tool Management System for Advanced Manufacturing", Manufacturing Technology International – Europe 1989, Sterling Publications Ltd.)	LUT–SE Research Group

Project Team: LUT-SE Research Group
Title: Figure 15.4 Mazak Machining Line Model
Project: Manufacturing Model
Printed by: A. Molina





A. Molina	Figure 15.5 Modelling the Rotational Parts Line	LUT–SE Research Group



A. Molina	Figure 15.6 Modelling the Rotational Parts Line Process Capabilities	LUT–SE Research Group

CELL	Facilities Strategies – Focus Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Product Group	OKP, **MP**, FMP, CP	If components require only turning process this Cell should not be used, instead use stand alone turning centres.	Variety of components Number of component Component Complexity
Product Volume	HV–HS, HV, LV, **LV–LS**	Operate on batches of minimum 1 job – full pallets maximum n jobs – full pallets	Cost per part fix regardless batch sizes Number of different components in a given time period
Product Life Cycle	**NP**, GP, **MP**, DP	New components require to be rotational with asymmetric features Priority can be given to manufacture components required from an extraordinary order. The rule is to schedule this job as soon as the material is available and the station required, within this Cell, is free.	Number of new components Lead times Cost of new component (if it is economical to be produced in the Cell)
Production Mode	P, **F**, **C**, L, Co	Use of Group Technology	Volume Lead time Cost per part
Product Group OKP: One of a Kind Products MP: Multiple Products FMP: Few Major Products CP: Commodity Products	Product Volume: LV–LS: Low Volume – Low Standardisation LV: Low Volume HV: High Volume HV–HS: High Volume – High Standardisation	Product Life Cycle: NP: New Products GP: Growing Products MP: Mature Products DP: Declining Products	Production Mode: P: Project (Job Shop) F: Functional (Batch) C: Cellular L: Line Co: Continuous Flow

Table 15.1. Description of the Facilities Strategies of the Rotational Parts Line

CELL	Capacity Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Product Range	HS, S, **LS**, **C**	Always increase the number of components which can be machined in the Cell (Now up to 200 different parts). Components can be made customized If new component can be machined at a suitable cost and time then do it.	Number of different components Cost per component Machining times
Cycle Demand	HE, HI, O, S	**If the capability is not enough to meet the lead times then .... Confidential Strategy**	Lead times Machine availability
Increase Capacity	O, IS, S, EF, AF	**If the capability is not enough to meet the deadlines then .... Confidential Strategy**	Volume Number of components in a given time Capacity utilization Space utilization
Product Range HS: Highly Standard S: Standard LS: Low Standard C: Customized	Cycle Demand: HE: Holding Excess Capacity HI: Holding Seasonal Inventories O: Overtime S: Subcontracting	Increase Capacity: O: Overtime IS: Increase Shifts S: Subcontracting EF: Enlarge Facility AF: Additional Facility in a new location.	

Table 15.2. Description of the Capacity Strategies of the Rotational Parts Line

CELL	Technology Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Processes	In Station: Turning, Milling, Boring, Drilling, etc. In Cell: Transportation	Type of Material: cast iron, steel and castings. If component only requires Turning Operations use stand alone machines	Cost Lead Time Tolerances
Resources	3 Stations with the following Components (Slant Turn 40N ATC Mill Center, Tooling System, Mazak Robot, 2–Pallet Changer). Auto Stacking Crane Pallet Stocker	Minimum Diameter 50 mm – Maximum Diameter 250 mm Minimum Length 15 mm Maximum Weight 45 K Keep the speeds and feeds under critical factors to avoid breakdowns.	Machine utilization Machining times Tool Life
Configuration	**HA**, PA, LA, M PS: **FMS – Three Stations** MHS: **Auto Stacking Crane** THS: **ATC** S/WS: **Pallet Stocker** SI: **DNC–Integration of Multiple FMCs**	Avoid the use of the ATC Keep the Pallet Stocker full	Flexibility ??
Layout	Intercell: **L**	Speed of Auto Stacking Crane is not important factor Auto Stacking Crane serves Stations and Pallet Stocker	Flexibility ??

Configuration (Degree of Automation):
HA: Highly Automated
PA: Partial Automated
LA: Low Automated
M: Manual

PS: Processing System (Operator–assisted NC machines, FMS, etc)
MHS: Material Handling System
THS: Tool Handling System
QAS: Quality Assurance System
S/WS: Storage/Warehouse System
SI : System Integration (Integration of Multiple FMC, Integration with CAD/CAM, etc.)

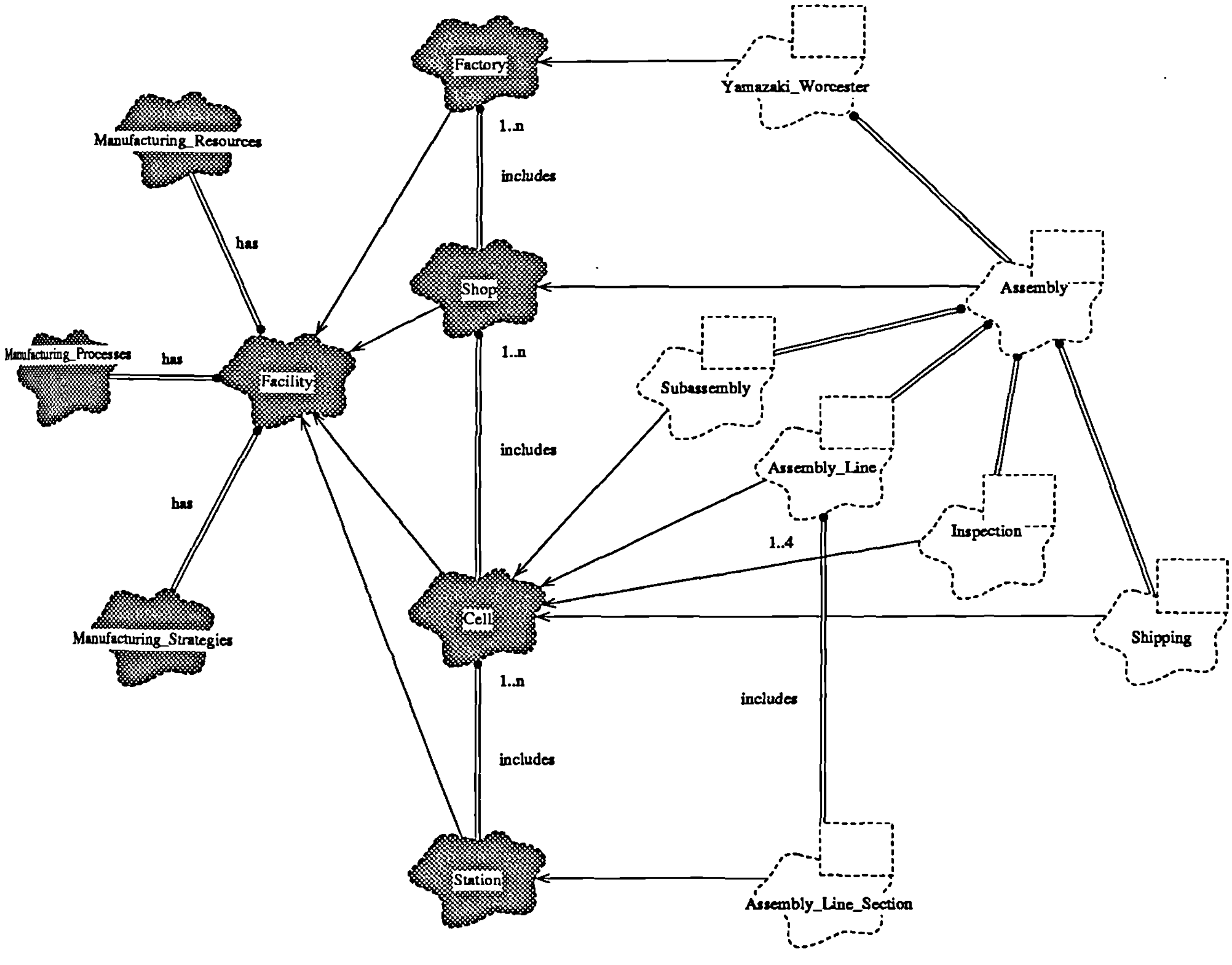
Layout (Intercell Type):
L: Linear Layout X: X Layout
T: T Layout
N: Network of Cells
CF: Common Facilities
CC: Cascading Cells.
PF: Parallel Flow–lines
VC: Virtual Cells

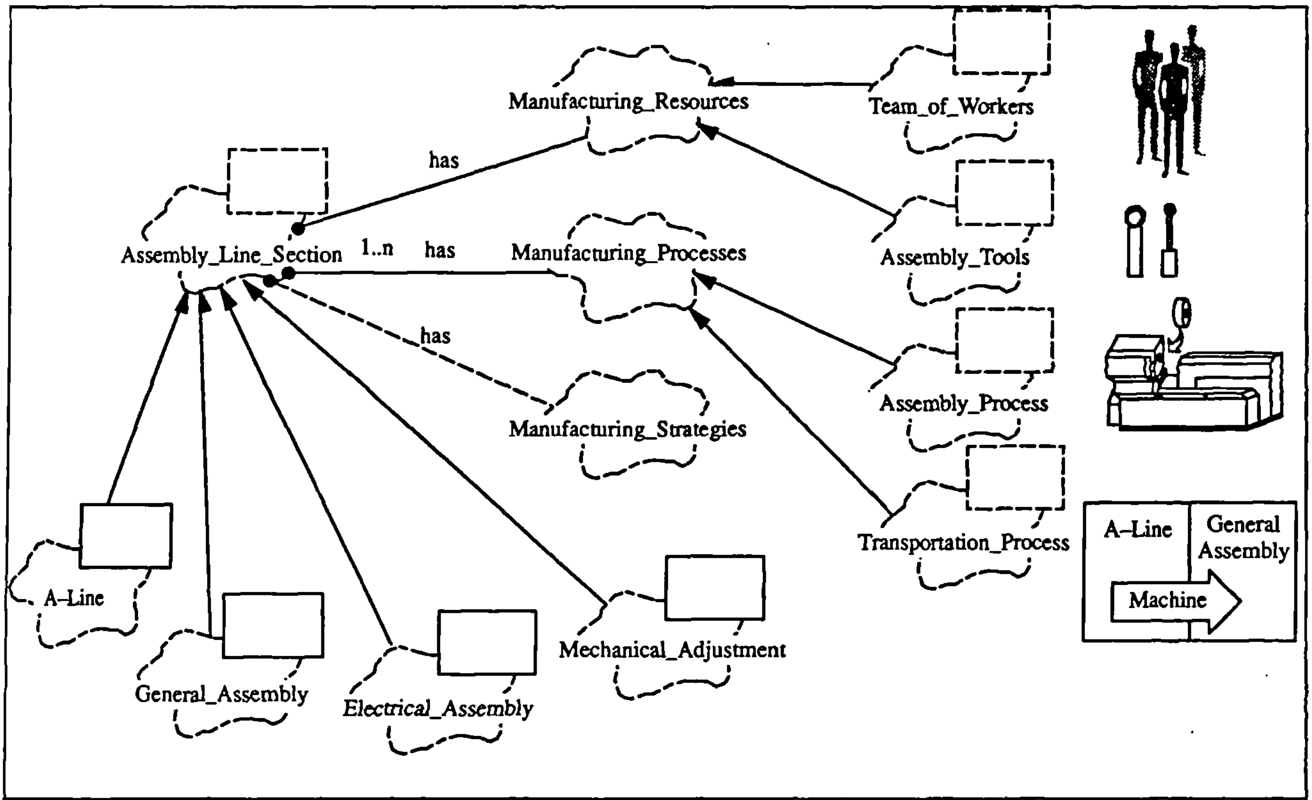
Table 15.3 Description of the Technology Strategies of the Rotational Parts Line

CELL	Production Planning/ Material Control Strategies		
	Strategic Decisions	Operational Rules	Performance Measures
Master Scheduling	ATO, DTO, **ETO**, **MTO**, MTS	The components produced in this cell are generally Make to Order Some components are machined as spares to be kept in the Warehouse for future demand The factory can produce special machines based on special customer configuration	Lead times
Shop Flow Control	Push Type **Pull Type**	Based on 5 lead times: A–Line Start, A–Line Release, Unit Release, Bases to Cast and Paint, and Unit Parts	Lead times
Material Planning	**Time–Phased** Rate–Phased	All the material is order as required from the Warehouse The material is order weekly	Lead times
Cell Scheduling	Group Technology allocation	Allocate job to the Station which match job requirements against: jaws (sizes and dimensions), tool set, and robot capabilities (size and dimensions). Flexible factor is the jaws. There are 3 carousels in each Station to meet the requirements of the component. If the tool set can no machine the job use another Station If the robot can not handle the job use another Station Use Station with Slant Turn 40N ATC Mill Center with work cross for large components (length > 80 mm)	Lead times Station utilization
Master Scheduling: ATO: Assemble To Order DTO: Design To Order ETO: Engineered to Order	MTO: Make To Order MTS: Make to Stock		

Table 15.4 Description of the Production Planning/Material Control Strategies of the Rotational Parts Line

Project Team:	LUT-SE Research Group
Title:	Figure 15.7 Assembly Shop Partial Model
Project:	Manufacturing Model
Printed by:	A. Molina





A. Molina	Figure 15.8 Modelling the Assembly Line Sections	LUT-SE Research Group

Chapter 16

Concluding Discussion

16.1. Introduction

The areas of work where this research has been focused resulted in a number of issues which have been brought together in this concluding discussion. This chapter allows a final balance of argument to be exercise in the work reported in this thesis. This discussion also indicates those issues which are important but were only explored superficially, such as: the representation of assembly process and real time support.

16.2. A Manufacturing Model to represent Batch Manufacturing

Manufacturing Modelling has at the centre of its philosophy the need to provide designers and manufacturing engineers with high quality manufacturing information on which to base their decisions (Young 1994). The provision of a source of core information provides a basis from which concurrent design and manufacture can be supported. The Manufacturing Model concept reported in this thesis describes and captures the information regarding the manufacturing facility of a particular enterprise in terms of its structure, organisation, and capabilities (5, 5.3). To achieve general applicability the model is comprised of entities that are relevant and important for any type of manufacturing firm, namely: resources, processes and strategies (11, 11.3). By combining these entities it is possible to represent generic and company specific manufacturing information (11, 11.4). These manufacturing entities have been organised in functional levels to achieve a generic representation which can be tailored to suit different enterprises (11, 11.5).

This research concentrates in the machining sector, in particular Flexible Manufacturing Systems (13, 14). By pursuing detailed modelling in this area, it has been showed both, the potential applicability of the Manufacturing Model as a source of dependable manufacturing information, and its readily use to the industry.

On the other hand, the author has been able to look inquiringly to some issues related to Batch Manufacturing, other than machining. It has been noted that the potential link of the Manufacturing Model to real time will require a significant effort by researchers who decide to follow this research (14). Finally, while studying the viability of modelling assembly based on the activities with Yamazaki, the author has been able to offer some opinions only rather than results on this subject (15).

16.3. The Manufacturing Model against other competitive information models

The Manufacturing Model presented in this thesis has common elements with other competitive models ISO TC184/SC4/WG8, CIMOSA (ESPRIT Project 688/5288), Kimura (Kimura 1991) and in particular IMPPACT (IMPPACT 1991). The work of WG8 has concentrated in representing only resources, but it has produced good guide–lines for the modelling work on this area. CIMOSA has been targeted not only to model information but the complete enterprise which includes the modelling of its dynamic behaviour. This represents a potential problem due to its complexity. However the concepts generated by CIMOSA have set the foundations for integrated modelling approaches. Kimura has defined representations of resources and processes using an object oriented approach. This approach has been found to be powerful and flexible. The closest competitive work is the one of IMPPACT, where a combined use of information models which represent resources and processes has been made. Nevertheless the issues related of how to represent strategies in a wider sense has been left out. Only operational strategies related to how to perform machining processes have been considered. The research reported in this thesis has been influenced by these competitive work in the following manner: use of guide–lines and generic classifications defined by WG8, the establishment of an integrated approach similar to CIMOSA, applications of object oriented methods comparable with Kimura's approach, and the combined representation of resources and processes, and limited modelling of cutting strategies employed by IMPPACT.

The Manufacturing Model concept presented here has contributed to further develop all these ideas by combining and establishing comprehensive associations between manufacturing resources, processes and strategies (11, 11.3). This integrated representation enables to capture generic manufacturing capability information, and allows the description of company's strategic decisions and operational rules (11, 11.4). The Manufacturing Model also provides the means to relate strategic and operational information by using a coherent and multi–level representation (11, 11.5).

It has to be noted that the Manufacturing Model has the characteristics of being modular and generic (11, 11.2). Therefore its partial use is possible. However the author would like to point out that although the development of a partial Manufacturing Model is simpler, the time and effort required to model manufacturing information has not to be underestimated.

16.4. The potential of the Manufacturing Model to support the Product Life Cycle

The Manufacturing Model is ultimately intended to be used to support design and manufacturing activities, i.e. the life cycle of a product (5, 5.5). This support will be possible by using both Product and Manufacturing Models which will provide the necessary information to all the activities of the product life cycle. It is accepted that a Product Model captures the information related to a product throughout its life cycle (Krause et al. 1993). Therefore the need to describe and capture the information regarding the manufacturing processes and resources required through the life cycle of a product is considered equally important by the author.

The manufacturing information used by many of the life cycle activities (e.g. design, manufacturing, assembly, testing) is similar but the interpretation placed on that information is what differentiates the range and detail of information needed for a particular activity. For example, the planning and generation of NC code needs to know the processes that can be undertaken by a given resource as does the design for manufacture activity. However the level of detail is different but not the basic information schema.

The Manufacturing Model can be considered a comprehensive schema which presents a common view of the resources, process and strategies within a company. If the life cycle is to be supported then multiple databases which hold the information represented in the schema will be required. Nevertheless some information which is application specific do not belong to the Manufacturing Model but to the application itself. An example of this case is the machining parameters defined for certain process (e.g. turning, milling) or the algorithms for the selection of tools and machining parameters, which the author believes should be kept in specialized databases or in the specific application.

16.5. The search for a formalism to support manufacturing modelling

One key issue in this research has been the search for a formalism to support manufacturing modelling. In this research a formalism has been used to set a context for the research in modelling the manufacturing information, establish a methodology for the development of the Manufacturing Model and assure the creation of a consistent, complete and sound Manufacturing Model (11, 11.2). In addition, the use of formal methods in a structured manner has enabled meaningful discussions to take place between the author and the researchers of the MOSES project and has allowed a common understanding of underlying goals and ideas.

In order to set the context for this research the author has explored and employed the concept of reference models (7, 7.2). Their importance and value are discussed later in section 16.6. A key problem is the selection of the method or set of methods to be used in order to capture all the important elements of the Manufacturing Model in a formal representation, and hence guide its development. These methods should be in accordance with the concepts of the reference model(s) employed (10, 10.2). To tackle this problem an integrated methodology to model manufacturing information has been defined in this thesis (11, 12). The author has conceived this methodology in order to facilitate the modelling work. The methodology is simple and adequate for the task of representing and capturing manufacturing information. One important aspect of the methodology is the use of two well known methods, i.e. EXPRESS language (10, 10.3.2) and Booch Object Oriented Methodology (10, 10.3.3) . The author believes that these methods are the most appropriate and better map the concepts introduced in the integrated methodology (12, 12.2).

The research reported in this thesis has caused the author to recognise the need for integrated methodologies. There is no single, integrated methodology which embraces all the important aspects of the modelling work. Therefore the use of different methodologies is evident. However the main problem of this approach is the translation between the results obtained from one methodology to the other. This is a major drawback because the gap between graphical and textual representations is considerable. Consequently it is important to analyse, match and select the methodologies which can *be most appropriate for the job* and *can be easily* integrated.

There is no best methodology as such, the best methodology is the one which fulfil the requirements of the person who employs it. The selection of a methodology should be based on three parameters: type of the task to perform, experience and availability of tools. On the other hand, if a methodology is employed within a reference model, then the methodology should be in accordance with the reference model guide–lines.

16.6. The importance and value of using Reference Models

It is important to select the right reference model(s) to assist in the realization of a task. In this research it has been recognised that a reference model was needed to: assist in the elicitation of requirements for CAE Systems, provide guide–lines to develop CAE Systems, and enable the organisation of methods and tools to introduce a systematic approach to CAE system development (7, 7.2).

Not all the reference models found in the literature are complete (Williams et al. 1993). Therefore a combined or partial use of them seems to achieve better results. This is the main reason behind the creation of the formalism which combines the use of CIMOSA and RM–ODP, called in this thesis the CAE Framework (7, 7.2). The second major reason to use this hybrid approach is related to the nowadays limitations of CIMOSA. In theory, it is possible to use CIMOSA to design and implement integrated CAE systems, nevertheless at this time the CIMOSA models for designing and implementing systems are not ready yet to be fully exploited (ESPRIT Project 688/5288).

The use of CIMOSA Requirements Definition Model solved the dilemma between the difference of opinion in the MOSES research project related to how to supply a specification for CAE systems (7, 7.3). This partial CIMOSA model represents the enterprise requirements for CAE systems to support simultaneous engineering (8, 8.2). This model is therefore a driver for the definition of the functions and capabilities required from a CAE system.

One major contribution of this research was to assist in the development of a Reference Model for future CAE systems (CAE–RM) in the MOSES project (5, 5.1). The CAE–RM describes the functionality, configuration and technology that is necessary to satisfy the requirements specified by the CIMOSA Requirements Definition Model. Therefore the MOSES project has used the RM–ODP model, which provides a suitable basis for the definition of the CAE–RM (9, 9.2). One important characteristics of RM–ODP is that its applicability is on the area of Framework and Reference Model developments (ISO/IEC 10746–1). Thus its documents are guide–lines for the development of further standards, and define the overall structure of systems. Based on these facts the use of RM–ODP was considered by the author to be suitable for this research (7, 7.4).

On the other hand, the use of RM–ODP to design and develop the CAE system is appropriate because RM–ODP is intended to be compatible with other open distributed system, in fact the CIMOSA integrating infrastructure has been developed in parallel with the RM–ODP with the best possible consistency (ESPRIT Project 688/5288).

16.7. The applicability of the Manufacturing Model

Experimental work of others researchers, which will be demonstrated later in the year, will use and verify that the Manufacturing Model prototype (Appendix K) is able to support this complementary research work. This future work will further validate the case for the Manufacturing Model. In order to demonstrate the use of the Station Level of the Manufacturing Model, two scenarios have been envisaged by the researchers of the MOSES project and the author: Design for Manufacture and NC Planning.

Two data–driven applications will be developed to exploit the Manufacturing Model concept. The research works of T.I.A. Ellis (Ellis 1995), research associated of the MOSES project working in the Design for Manufacture Strategist, and Dr. A. Tavakoli (Tavakoli 1993), a former PhD research student who developed a Feature Based Workshop Oriented NC Planning System for Asymmetric Rotational Parts, have been taken as the basis for the two demonstrations. In addition, the work in Design for Injection Moulding of R.J.V. Lee (Lee 1993) is exploiting the Manufacturing Model developed by Dr. A. Al–Ashaab (Al–Ashaab 1994).

16.8. Limited exploration of the role of the Manufacturing Model to support Real Time Control Applications

At the beginning of this research work the objective was the definition and development of a Manufacturing Model to support Design for Manufacture Applications (5, 5.3). During the conception of the Manufacturing Model the author recognized that the applicability of the model could be extended to support not only design activities but manufacturing activities as well (5, 5.5). Nevertheless because the extent of this topic, only key issues related to the support of manufacturing activities were identified and reported here (15, 15.6). The author believes that the results presented in this dissertation are solid foundations for the exploration of new research in this area.

The support of real time control applications requires three important aspects of the Manufacturing Model to be further developed. The first one is the representation of the behaviour dimension (11, 11.3.3) in a way that really represent the status and state of the manufacturing resources and processes through time. To enable this, there is a need to create a new level of instances, the ones called in the ISO/TC184 SC4 WG8/N9 Individual Resources (12, 12.3) which allow the representation of temporal characteristics such as material planning dates, wear conditions, status, process availability, etc. The second aspect to be considered is the integration of the Manufacturing Model with the so called real time databases which support real time operations. This will allow the Manufacturing Model to have ready access to the status and real time situation of the manufacturing facilities. The last aspect, but no less important, is regarding the way strategies should be implemented in a Manufacturing Model database. There is a need to find a dynamic representation of the strategies in order that they can be accessed and used by the applications to support the decision making.

16.9. Limited exploration of the representation of the Assembly Process

The work on assembly has followed the approach of decomposing a product into components and associated to each of them an assembly task, in this way it is possible to define the capability of the process and related resources employed in this task (Boothroyd and Alting 1992). This approach represents a potential problem to the idea of representing a Manufacturing Model which is product independent (5, 5.3). However this representation allows the characterisation of the assembly capability in terms of how many products can be assembled in a given period, time required for each product, processes and resources required and costs (15, 15.7). Perhaps there is a need to relax the conception of a product independent Manufacturing Model and introduce a special case for the assembly process representation. In the exploratory exercise carried out in this thesis the assembly process was defined in terms of its inputs/outputs/controls and this definition was associated to the facility where such a process can be performed, i.e the assembly line and its subsections. This representation defines in a limited manner the capability of an assembly process within the factory. Nevertheless the author considers that the main problem with this representation is that huge Manufacturing Models can be generated, if the Manufacturing Model is required to keep the associations of the assembly facility to the type of products can be assembled on it. Therefore further research is needed in this area.

16.10. The value of the Manufacturing Model to the industry

For some time it has been recognized that information is a major asset in a company (Scheer 1989). The work on representing product related information, especially geometry definition, is relatively mature and the evolving STEP standard is on its way to take a place in the industrial practice (ISO CD 10303 – 1). The emphasis of information modelling is therefore shifting to include models other than those that represent product data, such as factory models, that include resources and/or material flow models (Krause et al. 1993). Manufacturing modelling is a new research area which intends to address the problem of representing and capturing manufacturing information. The research reported in this thesis has addressed especially the issue of modelling the capabilities of manufacturing facilities (5, 5.3). This work will have a short and long term impact in the industry. In the short term, it will influence the way of how industry can standardise, organize and structure their databases which support contemporary engineering practices such as resource management (13, 13.4). As for the future value to the industry, it can be predicted that the next generation of Computer Aided Engineering Systems will required the use of the Manufacturing Model to capture and describe one important element of the enterprise information, i.e. the manufacturing information.

16.11. Concluding Remarks

This chapter has been written in order to provide a balance of argument based on the research reported in this thesis from which the reader will be able more readily to appreciate the background of the conclusions reported in the next chapter. There is not a one to one mapping between the broad headings of discussion in this chapter to the conclusion. It should have to be noted that the first five conclusions are considered to be the major conclusions of the thesis and the remainder are significant supporting issues.

Chapter 17

Conclusions and Further Research

17.1. Conclusions

- A competitive and novel Manufacturing Model concept has been defined. This concept has been researched using Flexible Manufacturing Facilities instances in order to represent their manufacturing capability giving particular attention to the machining processes.
- This Manufacturing Model is based on a set of original concepts in the modelling of manufacturing information, which the author believes are in advance of those reflected in similar research work found in the literature.
- The new concepts which have been presented include: integrated representation of generic and company specific information, establishment of an understandable relationship between strategic and operational information, and the association of three key manufacturing information elements, i.e. resources, processes and strategies.
- The Manufacturing Model concept emerged from this research work offers a considerable potential to support the product through its life cycle. The concept of using both Manufacturing Model and the Product Model to give continuous, high integrity, support to the activities which go beyond that of design i.e. manufacturing information generation, manufacturing, assembly, testing, maintenance and disposal.
- Considerable importance has been given in this research to establish a formal basis for generating Manufacturing Models. A formalism has been constructed and as a consequence a powerful Integrated Modelling Methodology has been created.

- The Integrated Modelling Methodology encompasses an integrated approach to model the relevant elements of any manufacturing environment (i.e. resources, processes and strategies) and a structured schema to represent strategic and operational manufacturing information. This methodology also utilizes novel techniques for information modelling which are becoming standards i.e. the object oriented methodology Booch and the international standard language for information modelling EXPRESS.

- The Manufacturing Model has been conceived in a structured and modular manner. Therefore some elements of the Manufacturing Model have an immediate applicability for industrialists.

- The elements of the work which have been identified as being particular relevant to industry are the generic characterizations of resources, process and strategies; these offer a readily applicable foundation for the design and implementation of industrial databases. The case studies presented in this thesis have proven this point in the form of a prototype Manufacturing Model which will support the following applications: Design for Manufacture, Process Planning and NC–Code generation.

- The use of reference models has been found to be a helpful concept to allow a systematic approach. It does however offers challenges to the users due to the complexity of producing complete models. Nevertheless partial models can be built and they are simple to deal with and very effective.

- The use of reference models in general however does present a major challenge to industrial users, but it is not essential to produce complete reference models in order to gain considerable value for using them in particular areas of relevance to the industry such as: system integration and enterprise modelling.

17.2. Further Research

Two issues of the future usability of the Manufacturing Model have been identified as important but have been not completed explored. These issues are related to the role of the Manufacturing Model to support real time control applications and the representation and modelling of the assembly process and other manufacturing process in a manner which can be useful to realize engineering activities.

When one considers the use of the Manufacturing Model to support product life cycle then an obvious area of relevance is that of manufacturing leading to the support of real time control issues. The research centred on the specific application area of the lower levels of the Manufacturing Model and only a limited reference could be made to assembly. Further work on the Manufacturing Model should be devoted to assembly and should be given a high priority.

REFERENCES

ACME/SERC Proposal, 1991, "Exploiting Product and Manufacturing Models in Simultaneous Engineering", Department of Mechanical Engineering–University of Leeds, Department of Manufacturing Engineering–Loughborough University of Technology. UK.

Adiga S., 1993, *Object–Oriented Software for Manufacturing Systems*, Edited by S. Adiga, Chapman & Hall, London.

AFNOR, 1989, SET, Z 68–300–1, ISSN 0335–3931, Published by AFNOR, France, June.

Al–Ashaab A., and Young R.I.M, 1992, "Information Models: An Aid to Concurrency in Injection Moulded Products Design", *Presented in the Winter Annual Meeting '92 of ASME*, Anaheim, California, November 8–13.

Al–Ashaab, A.H., 1994, "A Manufacturing Model to Capture Injection Moulding Process Capabilities to Support Design for Manufacture", PhD Thesis, Manufacturing Engineering Department, Loughborough University of Technology.

Albus J.S., Barbara A.J., Nagel R.N., 1981, "Theory and practice of hierarchical control", *In Proceedings of the IEEE Computer Society International Conference.*

Allen D.K., 1987, "An Introduction to Computer–Aided Process Planning", *CIM Review*, Fall, pp. 7–23.

Als Pedersen M., and Alting L., 1991, "Life–Cycle Design of Products – An Important Part of the CIM Philosophy", *AUTOFACT'91*, Conference Proc., November 10–14, 1991, Chicago, Illinois, SME, pp. 2:1–2:16.

Alting L., 1991, "Life–cycle Design of Products, A new opportunity/challenge for manufacturing enterprises", *Concurrent Engineering: Issues, Technology and Practice*, November/December.

Alting L., and Jorgensen J., 1993, "The Life Cycle Concept as a Basis for Sustainable Industrial Production", *Annals of the CIRP*, Vol. 42/1/1993, pp. 163–167.

Alting L., and Zhang H., 1989, "Computer Aided Process Planning: the state–of–the–art survey", *Int. J. Prod. Res.*, Vol. 27, No. 4, pp. 553–585.

Alting, L., 1982, *Manufacturing Engineering Processes*, Marcel Dekker, Inc. New York and Basel.

Amble T., 1987, *Logic programming and knowledge engineering*, Addison–Wesley.

Apple, J.M., 1972, *Material Handling Systems Design*, New York: The Ronald Press Company.

Appleton, 1985, *Information Modeling Manual–EDF1 Extended (IDEF1x)*, D. Appleton Co., December.

Arvindh, B., and Irani S.A., "Cell formation: the need for an integrated solution of the subproblems", *International Journal of Production Research*, 1994, Vol. 32, No. 5, 1197–1218.

Banerjee S.K., and Al–Maliki L., 1992, "A structured approach to FMS modelling", *Int. J. Computer Integrated Manufacturing*, Vol. 1, No. 2, 77–88.

Barret J., 1994, "Concurrent Engineering – A rapid product development plan", *Engineering Computers*, Vol. 13, No. 2, April, pp. 16–90.

Bauert F., 1989, "Composition of product modelling systems based on commercial software and their influence on industrial companies, design methods and education", *Proceedings of the Institution of Mechanical Engineers*, International Conference, Engineering Design, Volume 1, pp. 655–676.

Baumgarater H., Knischewski K., and Widing H., 1989, Basic CIM Considerations, *Siemens Akteingesellschaft*, Produktionsautomnatisterung und Automatisierungs –systeme, Produktionsleittechnik, Nürnberg–Moorenbrun (in German).

Bedworth, D.D., Henderson M.R., and Wolfe P.M., 1991, *Computer–Integrated Design and Manufacturing*, McGraw–Hill, Inc., International Edition.

Bernus P., and Nemes L., 1994, "A Framework to Define a Generic Enterprise Reference Architecture and Methodology", Draft proposal, *IFAC/IFIP Task Force on Architectures for Enterprise Integration*, Eight Workshop Meeting, Vienna, Austria, June 11–12, 1994.

Bernus P., Nemes L., and Morris R., 1994, "Possibilities and Limitations of Reusing Enterprise Models (New Requirements for Enterprise Engineering Tools)", Preprints, *2nd IFAC/IFIP/IFORS Workshop Intelligent Manufacturing Systems – IMS'94*, (Ed. P. Kopacek), Vienna – Austria – June 13–15, 1994, pp. 11–16.

Berry, W.L and Hill, T., 1992, "Linking Systems to Strategy", *International Journal of Operations & Production Management*, Vol 12, No. 10, pp. 3–15.

Besant, C.B., and Lui, C.W.K., 1989, *Computer–Aided Design and Manufacture*, Third Edition, Ellis Horwood Limited, Bey I., 1989, "CAD Interface, a key to Computer Integrated Manufacturing", *Conference Proceeding ESPRIT 1989*, pp. 31–45.

Bey I., and Gengenbach U., 1988, "The CAD*I Interface for Solid Model Exchange", *Computer and Graphics*, Vol. 12, No. 2, pp. 181–190.

Biocca F., 1992, "Virtual Reality Technology: A Tutorial", *Journal of Communication*, Volume 42, Issue 4, pp. 23–73.

Bishop R., 1991, "Concurrent engineering practices: a matching of cultures speeds time to market", *Eureka*, September, 11(9), pp. 102–111.

Bjorke O., and Myklebust O., 1992, *IMPPACT: Integrated Modelling of Products and Process using Advanced Computer Technologies*, TAPIR Publishers, Trondheim, Norway.

Blinn T., Mayer R.J., and Cullinane T., 1991, "A *Service Based Approach to* Information Integration in Concurrent Engineering Environments", *AUTOFACT'91*, Conference Proc., November 10–14, 1991, Chicago, Illinois, SME, pp. 7:27–7:38.

Bloor M.S., De Pennington A., Harris S.B., Holdsworth D., Mckay A., and Shaw N.K., 1988, "Toward Integrated Design and Manufacturing", *International Conference on Factory 2000*, 31 August – 2 September, Churchill College, Cambridge, pp. 21–29.

Bloor M.S., and Owen J., 1991, "CAD/CAM product–data exchange: the next step", *Computer Aided Design*, Vol. 23, No. 4, May, pp. 237–242.

Booch G., 1991, *Object–oriented design with applications*, Benjamin/Cummings Inc., 1991.

Boothroyd D., 1985, *Design for Assembly Handbook*, Boothroyd Dewhurts, Inc., Wakefield.

Boothroyd, G., and Alting, L., 1992, "Design for Assembly and Disassembly", *Annals of the CIRP*, Keynote Paper, Vol. 41/2/1992, pp. 625–636.

Booty F., 1991, "Approaches to Design for Manufacture", *Works Management*, June 44(6) pp. 57,59,61.

Bowen J., and Bahler D., 1993, "Constraint–Based Software for Concurrent Engineering", *IEEE Computer*, 26(1), pp. 66–68.

Bowen J., Bahler D., and Dholakia D., 1993, "An AI Constraint Network–Based Approach to Bed–Of–Nails DFT for Digital Circuit Design", *Computers Elect. Engineering*, 19(2), pp. 73–85.

Bravoco R.R., and Yadav S.B., 1985a, "Requirement Definition Architecture–An Overview", *Computers in Industry*, Vol. 6, pp. 237–251.

Bravoco R.R., and Yadav S.B., 1985b, "A Methodology to Model the Information Structure of an Organization", *The Journal of Systems and Software*, Vol. 5, pp. 59–71.

Bravoco R.R., and Yadav S.B., 1985c, "A Methodology to Model the Functional Structure of an Organization", *Computers in Industry*, Vol. 6, pp. 345–361.

Bravoco R.R., and Yadav S.B., 1985d, "A Methodology to Model the Dynamic Structure of an Organization", *Information Systems*, Vol. 10, No. 3, pp. 299–317.

Bray O.H., 1988, *CIM–The Data Management Strategy*, Digital Press.

Bruce R.G., 1990, "The CAM Journey from CAD to CNC: Numerical Control Postprocessing", *Proceedings of the 1990 ASEE National Conference*, Toronto, Ontario, Canada, ASEE, June.

BSI DD 194:1990, ENV 40 003:1990, Computer integrated manufacturing (CIM): CIM systems architecture framework for modelling, *British Standard Institution.*

BSI DD 203: Part1 : 1991 ISO/TR 10314–1 :1990, Industrial automation shop floor production, Part 1. Reference model for standardization and a methodology for identification of requirements, *British Standard Institution.*

BSI PD 6526:1990, Evaluation report on CIM architectures, *British Standard Institution*, CEN/CENELEC Report R–IT–01.

Bunce P., Pratt M.J., Pavey G., and Pinte J., 1986, "Features Extraction and Process Planning", *CAM–I Report R–86–PPP–01*, Arlington, Texas.

Burkett W.C., and Yang Y., 1992, "On the implementation of STEP", *STEP Documentation*, October, P.D.I.T., Inc. 3780 Kilroy Airport Way Suite 430 Long Beach CA 90806.

Busby J.S., and Williams G.M., 1993, "The value and limitations of using process models to describe the manufacturing organization", *Int. J. Prod. Res.*, Vol. 31, No. 9, 2179–2194.

Butterfield W.R., Green M.K., Scott D.C., and Stoker W.J., 1986, "Part Feature for Process Planning", *CAM–I Report R–86–PPP–01*, Arlington, Texas.

Byrd J., Wood R., and Farquer A., 1992, "Overcoming Resistance to IPD/CE", *CE & CALS Conference and Exposition*, June 1–4, Washington, D.C., pp. 521–535.

Byrne G., and Scholta E., 1993, "Environmentally Clean Machining Processes – A Strategic Approach", *Annals of the CIRP*, Vol. 42/1/1992, pp. 471–474.

CADDETC, 1990, Initial Graphics Exchange Specification (IGES) Version 5.0, NISTIR 4412, *Published by CADDETC*, Leeds, England, September.

CAM–I, 1989, "Functional requirements for a Feature Based Modelling System", *CAM–I–R–89–GM–01*, Arlington, Texas.

Case K., and Gao J., 1991, "Feature Technology – An Overview", *Seminar on Feature–Based Design*, Loughborough University of Technology, pp. 1–14.

Case, K., Gao, K., and Gindy N., 1993, "Feature Validation in a Feature–Based Design System", In Advances in Manufacturing Technology, A. Bramley and T. Mileham (eds.), Proceedings of the Ninth National Conference on Manufacturing Research, University of Bath, September 1993, COMED.

CFI 1991, *CAD Framework Initiative*, Tool Encapsulation Specification, April 17.

Chadha B., Jazbutis G., Wang C–Y., and Fulton R.E., 1991, "An Appraisal of Modeling Tools and Methodologies for Integrated Manufacturing Information Systems", In Engineering Databases: An Engineering Resource, *Proceedings of the ASME International Computer in Engineering Conference & Exposition*, (ed.) V. Saxena, August 18–22, Santa Clara, California, pp. 55–63.

Chang et al., 1988, "QTC–an integrated design and manufacturing/inspection system for prismatic parts", *Proceedings of the 1988 ASME Computers in Engineering Conference*, San Francisco, USA, 1–4 August.

Chang, T.C., and Wysk R.A., 1984, "Integrating CAD and CAM through Automated Process Planning", *International Journal of Production Research*, 22 (5), pp. 877–984.

Chen P.P., 1976, "The Entity–Relationship Model – Toward a unified view of data", *ACM Transactions on database systems*, Vol. 1, No. 1, pp. 9–36.

Chen P.P., 1979, "Entity–Relationship Approach to System Analysis and Design", *Proceedings of the International Conference on Entity–Relationship Approach to Systems Analysis and Design*, December 10–12, Los Angeles, California.

Chen Y., Dong Y., Zhang W., and Xie B., 1994, "Proposed CIM Reference Architecture", Preprints, 2nd IFAC/IFIP/IFORS Workshop Intelligent Manufacturing Systems -- IMS'94, (Ed. P. Kopacek), Vienna – Austria – June 13–15, 1994, pp. 47–50.

Choi B.K., Barash M.M., and Anderson, 1984, "Automatic Recognition of Machined Surfaces from a 3D solid model", *Computer Aided Design*, Vol. 16, No. 2, pp. 81–86.

Cleetus K.J., 1992, "Definition of Concurrent Engineering", CERC Technical Report Series, *CERC–TR–RN–92–003*.

Coad, P., and Yourdan E., 1990, *Object–oriented analysis*, Prentice–Hall Inc.

Colby E., 1990, "Environmental Management in Development", World Bank Discussion Paper 80, Washington D.C.

Coleman D.S., 1989, "A Framework for Characterizing the Methods and Tools of an Integrated System Engineering Methodology (ISEM)", *Pacific Information Management*, Draft 2 Rev. 0, May 1989, p. 2

Colquhoun, G., Baines, R.W., and Crossley R., 1993, "A state of the art review of IDEF0", *Int. J. Computer Integrated Manufacturing*, Vol. 6, No. 4, 252–264.

Conkol G.K., 1990, "The Role of CAD/CAM in CIM", *AUTOFACT'90*, Conference Proc., November 12–15, 1990, Detroit, Michigan, SME, pp. 8:1–8:16.

Conkol G.K., 1991, "The Role of CAD/CAM in CIM – Part II", *AUTOFACT'91*, Conference Proc., November 10–14, 1991, Chicago, Illinois, SME, pp. 6:1–6:15.

Corbett J., Donner M., Meleka J., and Pym C. (Eds.), 1991, *Design for manufacture, strategies, principles and techniques*, Addison–Wesley Publishing Company.

Corrigall M.J., Lee M.K., Young R.I.M., and Bell R., 1992, "Manufacturing Code Generation in a Product Modelling Environment", *Proc. Instn Mech. Engrs.*, Vol. 206, pp. 165–175.

Craig N., 1991, "Why Concurrent Engineering Programs Don't Create Concurrent Engineering", *AUTOFACT'91*, Conference Proc., November 10–14, 1991, Chicago, Illinois, SME, pp. 2:17–2:24.

Cunningham J.J., and Dixon J.R., 1988, "Designing with Features: The Origin of Features", *Proceedings 1988 ASME Int. Computers in Engineering Conference*, San Francisco, CA, pp. 237–243.

Cutkosky M.R., Brown D.R., and Tenenbaum J.M., 1989, "Extending Concurrent Product and Process Design Toward Earlier Design Stages", Concurrent Product and Process Design, N.H. Chao and S.C.Y Lu (Eds.), *The Winter Annual Meeting of the ASME*, San Francisco, California, pp. 65–72.

Cutkosky M.R., Engelmore R.S., Fikes R. E., Genesereth M. R., Gruber T.R., Mark W.S., Tenenbaum J.M. and Weber J.C., 1993, "PACT: An Experiment in Integrating Concurrent Engineering Systems", *IEEE Computer*, January, pp. 28–37.

Czernik S., and Quint W., 1992, "Selection of methods, techniques and tools for system analysis and for the integration of CIM–elements in existing manufacturing organizations", *Production Planning & Control*, Vol. 3, No. 2, pp. 202–209.

Dallas, D.B, Editor–in–Chief, *Tool and Manufacturing Engineers Handbook*, Third Edition, SME, 1976, McGraw–Hill Book Company.

Davies M., 1991, "The Status and Future of Product Data Exchange", *Proceedings Int. Conf. on Computer Integrated Manufacturing (ICCIM'91)*, October 2–4, 1991, Singapore, World Scientific, pp 259–262.

DeMarco T., 1979, *Structured Analysis and System Specification*, Yourdon Press – Prentice Hall, Englewood Cliffs, NJ.

Denton D.K., 1990, "QFD and Taguchi methods enhance productivity and quality", *Industrial Engineering*, May, 22(5), pp. 24–30.

Devgun M.S., and Padmanabham S., 1991, "Feature representation and extraction in CAD models for manufacturing: methods and issues", Production Research: Approaching the 21st Century, M. Pridham and C. O'Brien (Eds.), Taylor and Francis, pp. 749–763.

DiCesare F., Harhalakis G., Proth J.M., Silva M., and Vernadat F.,1993, *Practice of Petri Nets in Manufacturing*, Chapman & Hall, UK.

DIN Technical Report No. 15, 1988, The Standardization of Interfaces for Computer–Integrated Manufacturing (CIM) – Current state of development and future requirements, *Prepared by the DIN Commission for Computer–Integrated Manufacturing (KCIM)*, 1st edition, Beuth Verlag GMBH, Berlin, Koln.

Dixon, J.R., 1986, "Artificial Intelligence and design: a mechanical engineering view", *Proceedings of AAAI–86 Fifth National Conference on Artificial Intelligence*, Vol. 2, Philadelphia, PA, August, pp. 872–877.

Doumeingts G., Erkes K., Krezinski P., 1986, "Integration of Decisional and Physical Systems in the Design of Manufacturing Systems", *Proceedings of the IEEE 25th Conference on Decision and Control*, Athens, Greece, December, pp. 2212–2217.

Doumeingts G., Vallespir B., Darricar D., and Chen D., 1992, *GIM, GRAI Integrated Methodology, A Methodology for Designing CIM Systems*, Version 1.0, Unnumbered Report, LAP/GRAI, University Bordeaux 1, Bordeaux, France, May.

Doumeingts G., Chen D., Vallespir B., and Fenie P., 1993, "GIM (GRAI Integrated Methodology) and its evolutions – A methodology to design and specify Advanced

Manufacturing Systems", *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing, DIISM'93*, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 101–117.

Dowlatshahi, S., 1985, "A modelling approach to design of integrated facilities", *International Journal of Production Research*, 1994, Vol. 32, No. 6, 1313–1330.

Doyle, L. E., *Manufacturing processes and materials for engineers*, contributing authors , Carl A. Keyser ... (et al.), 3rd ed, Prentice–Hall.

Drake S., and Sela S., 1989, "A Foundation for Features", *Mechanical Engineering*, January, pp. 66–73.

Drozda T.J., and Wick C.,1989, *Tool and Manufacturing Engineers Handbook*, Four Edition, Volume I Machining, SME, McGraw–Hill Book Company

Dunn J., 1990, "Simultaneous Engineering", *Engineer 11*, October, 271 (7020), pp. 51–53.

Eastman C.M., 1991, "The Contribution of Data Modeling to the Future Development of CAD/CAM Databases", In Engineering Databases: An Engineering Resource, *Proceedings of the ASME International Computer in Engineering Conference & Exposition*, (ed.) V. Saxena, August 18–22, Santa Clara, California, pp. 49–54.

Ebach H., Scherer H., and Streibett H., 1992, "Recycling considerations in product life cycle", *Siemmens report ZPL1FPS14*, Erlangen, 1992.

ECMA, 1990, Support Environment for Open Distributed Processing (SE–ODP), *ECAM TR/49*, January.

Eirich P., 1992, "Enterprise Modelling: Issues, Problems & Approaches", *Reports of the ICEIMT Special Interest Groups*, June 6–10, 1992, Hilton Head, South Carolina.

EIS, 1990, Engineering Information Systems Specifications in 3 volumes: Volume 1: Organization and Concepts, October 1989; Volume 2: Specifications and Guidelines, October 1989; Volume 3: Engineering Information Model, September, 1990.

ElMaraghy, H.A., 1993, "Evolution and Future Perspectives of CAPP", Keynote Paper, Annals of the CIRP, Vol. 42/2/1993, pp. 739–751.

El Wakil, 1989, S.D., *Processes and Design for Manufacturing*, Prentice–Hall International Editions.

Ellis T.I.A., 1995, Untitled PhD Thesis, Manufacturing Engineering Department, Loughborough University of Technology.

ESPRIT Consortium AMICE, *Open System Architecture for CIM*, Research Reports of ESPRIT Project 688, Volume 1, Springer–Verlag, Berlin, 1989.

ESPRIT Project 688/5288, AMICE, Volume 1, *CIMOSA: Open System Architecture for CIM*, ESPRIT Consortium AMICE (Eds.), 2nd, revised and extended edition, Springer–Verlag, 1993.

Evans S., 1990, "Implementation framework for Integrated design teams", Journal of Engineering Design, 1(4), pp. 355–363.

Evans S., Lettice F., and Smart P., 1994, "Concurrent Engineering – Key Implementation Issues", *In Advances in Agile Manufacturing*, P.T. Kidd nad W. Karwowski (Eds.), IOS Press, 1994, pp. 89–92.

Eversheim W., and Gross M., 1991, "Trends and Experiences in Applying Simultaneous Engineering", *Proceedings of the Seventh CIM–Europe Annual Conference*, 29–31 May, Turin, Italy, pp. 3–11.

Eversheim W., and Shultz J., 1985, "Survey of Computer Aided Process Planning Systems", *Annals of the CIRP*, Vol. 34/2, pp. 607–613.

Eversheim W., Marczinski G., and R. Cremer, 1991, "Structured Modelling of Manufacturing Processes as NC–Data Preparation", *Annals of the CIRP*, Vol. 40/1, pp. 429–432.

Eversheim W., Rozenfeld H., and Marczinski G., 1989, "Requirements on Interfaces and Data Models for NC–Data Transfer in View of Computer Integrated Manufacturing", *Annals of the CIRP*, Vol. 38/1, pp. 443–446.

Farin, G., 1989, *Curves and Surfaces for Computer–Aided Geometric Design*, 2d ed., Academic Press, San Diego.

Faux I.D., 1986, "Reconciliation of Design and Manufacturing Requirements for Product Description Data Using Functional Primitive Part Features", *CAM–I Report R–86–ANC/GM/PP–01*, Arlington, Texas.

Fine C.H., and Hax A.C., "Manufacturing Strategy: A Methodology and an Illustration", *INTERFACES* 15:6, November–December 1985 (pp. 28–46).

Finger S., and Dixon J.R., 1989, "A Review of Research in Mechanical Engineering Design. Part II: Representations, Analysis, and Design for the Life Cycle", *Research in Engineering Design*, 1:121–137.

Finger S., Fox M.S., Prinz F.B., and Rinderle J.R., 1992, "Concurrent Design", *Applied Artificial Intelligence*, 6:257–283.

Finkenwirth K., and Jansen H., 1989, "Reference Model fo CAD Systems", International GI–IFIP Symposium 89, Appendix, IPK, November 8–10, Berlin (West), FRG.

Flatau U., 1988, *CIM Architectural Framework*, Digital Equipment Corporation, Mainard, MA.

Fox M., 1992, "The TOVE Project Towards a Common–Sense Model of the Enterprise", *in Enterprise Integration Modelling: Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 310–319.

Gadh R., Hall M.A., Gursoz E. L., Prinz F.B, 1989, "Knowledge Driven Manufacturability Analysis from Feature–Based Representations", *ASME Winter Annual Meeting*, Symposium on Concurrent Product and Process Design, pp. 21–34.

Genesereth M.R., Huyn P.N., and Letsinger R., 1992, "Agent–based Framework for Concurrent Engineering Software", *CE & CALS Conference and Exposition*, June 1–4, Washington, D.C., pp. 135–150.

Gerhardt D.J., Hutchinson W.R., and Mistry D.K., 1991, " Design for manufacture and assembly: case studies and its implementation", *International Journal of Advanced Manufacturing Technology*, May, 6(2), pp. 131–140.

Gielingh W., 1990, "IMPPACT as a Basis for Concurrent Engineering", *CALS Europe Conference*, The Hague.

Gielingh W.F., and Suhm A.K. (Eds.), 1993, *IMPPACT Reference Model – An Approach to Integrated Product and Process Modelling for Discrete Parts Manufacturing*, Springer–Verlag.

Gillen D.J., and Fitzgerald E.M., 1991, "Expanding knowledge and converging functions", *Concurrent Engineering*, May, 1(3), pp. 20–28.

Gindy N.N.Z., Huang X., and Ratchev T.M., 1991, "Feature–Based Component Model for Computer Aided Process Planning Systems", *Seminar on Feature–Based Design*, Loughborough University of Technology.

Gindy N.N.Z., Ratchev T.M., and K. Case, "Process Capability Models for Equipment Selection", Ir. Advances in Manufacturing Technology, K. Case and S.T. Newman (eds.), *Proceedings of the Tenth National Conference on Manufacturing*

Research, Loughborough University of Technology, 5–7 September, 1994, Taylos & Francis, pp. 244–248.

Goossenaerts J., 1993, "Enterprise Formulae and Information Infrastructures for Manufacturing", *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 123–140.

Goranson H.T., 1992, "Dimensions of Enterprise Integration", *in Enterprise Integration Modelling: Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 101–113.

Grabowski H., Anderl R., Holland–Letz V., Patzold B., and Suhm A., 1989, "An Integrated CAD/CAM–System for Product and Process Modelling", *Preprints of the International GI–IFIP Symposium of Advanced Geometric Modelling for Engineering Applications*, Production Technology Centre Berlin, November 8–10, Berlin (West), FRG, pp. 389–95.

Graefe U., and Chan A.W., 1993, "An Enterprise Model as a design tool for information infrastructure", Information Infrastructure Systems for Manufacturing, *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 183–192.

Groover M.P., 1987, *Automation, Production Systems, and Computer Integrated Manufacturing*, Prentice–Hall, Inc., Englewoog Cliffs, New Jersey 07632.

Groover M.P. and Zimmers E.W. Jr., 1984, *CAD/CAM: Computer–Aided Design and Manufacturing*, Prentice/Hall International, Inc.

Ham I., and Lu S.C–Y., 1988, "Computer Aided Process Planning: The Present and the Future", *Annals of the CIRP*, Vol. 37/2, pp. 591–601.

Hambaba M., Elsherif H.E., O'Brien W. and Bergstorm E, 1992, "Mold–base part Design Using Neural Networks", Intelligent Manufacturing System: *preprints of IFAC workshop*. (Eds. R.P. Judd, N.A. Kheir) Oxford, Pergammon, Dearbon, Michigan USA October, pp. 37–41.

Harding, J.A., and Popplewell K., 1994, "Engineering Moderation within a Concurrente Engineering Environment", Submitted to International Journal of Production Research, November.

Harhalakis G., Ssemakula M.E., and Johri A., 1992, "Functional design of an integrated CIM System at the facility level", *Int. J. Computer Integrated Manufacturing*, Vol. 1, No. 4, 245–252.

Hars A., and Scheer A.W., 1991, "From CIM to Enterprise–Wide Data Modelling", *Proceedings Int. Conf. on Computer Integrated Manufacturing (ICCIM'91)*, October 2–4, 1991, Singapore, World Scientific, pp. 89–92.

Hars A., Heib R., Kruse Chr., Michely J., and Scheer A.W., 1992, "Reference Models for Data Engineering in CIM", Proceedings of the Eight CIM–Europe Annual Conference, May 27–29, Birmingham, UK, Springer–Verlag, pp. 249–260.

Hartley, J., 1984, *FMS at Work*, Co–published by IFS (Publications) Ltd., UK and North Holland (a division of Elsevier Science Publishers BV).

Hartley J., 1990, *Simultaneous Engineering: the Management Guide*, Industrial Newsletters Ltd., Dunstable, UK.

Hayes R.H., and Wheelwright S.C., 1984, *Restoring Our Competitive Edge*, John Wiley & Sons, New York.

Henderson M.R., 1984, "Extraction of Feature Information from Three Dimensional CAD Data", Ph.D. Thesis, Purdue University, West Lafayette, Indiana.

Henderson M.R., 1986, "Automated group technology part coding from a three–dimensional CAD database", *Proceedings of ASME Winter Annual Meeting*, Knowledge–Based Expert Systems in Manufacturing, December, Anaheim, CA, pp. 195–204.

Henderson M.R.. and Anderson D.C., 1984, "Computer Recognition and Extraction of Form Features: A CAD/CAM Link", *Computers in Industry*, Vol. 6, No. 4, pp. 315–325.

Henshall, E.J., 1989, "Quality engineering", *TQM Magazine*, November, 1 (5), pp 293–298.

Hiatt D., 1990, "Designing for manufacturability: multifunctional teams at Hewlett–Packard", *CIM Review*, 6(4), pp 63–65.

Hofer C.W., 1978, and Schendel D., *Strategic Formulation: Analytical Concepts*, West Publishing Company, St. Paul.

Hon B., 1992, "Development of Concurrent Engineering in the USA", IFAC workshop, *SERC/IMechE Annual Expert Meeting*, Information Technology and

Product Design, Volume:1 Programme and Conference Volume, 8–10 September, Sopwell House, S Albans, England, pp. 116–123.

Horne, C.A., 1987,"Product Strategy and the Competitive Advantage", *P&IM Review with APICS News*, 7, no. 12, (December 1987): 38–41.

Hummel K.E., and Brooks S.L., 1986, "Symbolic representation of manufacturing features for an automated process planning system", *Proceedings of ASME Winter Annual Meeting*, Knowledge–Based Expert Systems in Manufacturing, December, Anaheim, CA, pp. 233–243.

Hummel K.E., and Brown C.W., 1989, "The Role of features in the Implementation of Concurrent Product and Process Design", Concurrent Product and Process Design, N.H. Chao and S.C.Y. Lu (Eds.), *The Winter Annual Meeting ASME*, San Francisco, CA, pp. 1–8.

Iacocca Institute, 1991, *21st Century Manufacturing Enterprise Strategy.* An Industry –Led View. Volumes 1 & 2. Iacocca Institute, Bethlehem, PA.

IBM, 1989, *Computer Integrated Manufacturing, The CIM Enterprise*, IBM Corporation, White Plains, N.Y.

ICEIMT Workshop I, 1992, The Notion of a Model: Report of Workgroup 1, in Enterprise Integration Modelling: *Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 17–22.

ICEIMT Workshop IV, 1992, Metrics and Models: Report of Workgroup 1, in Enterprise Integration Modelling: *Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 78–84.

IDS, 1989, "Integrated Design Support System (IDS) AFHRL–TR–89–6: Volume I – Executive Overview; Volume II – IDS Introduction and Summary; Volume III – IDS Requirements; Volume IV – IDS Task Results; Volume V – IDS Software Documentation. AFHRL, WPAFB, OH, December, 1989.

IMPPACT, 1991, "Esprit No. 2165", *Proceedings of the Workshop*, Berlin, 26–27 February 1991.

Industry–University Consortium, 1992, "An Implementation Procedure Manual for Developing Master Plans for Computer Integrated Manufacturing", *Report Number 155*, Purdue Laboratoy for Applied Industrial Control, Purdue University, West Lafayette, IN, June.

Inui M., Shuzuki H., Kimura F., and Sata T., 1986, "Generation and Verification of Process Plans Using Dedicated Models of Product in Computers", Knowledge–based Expert System for Manufacturing, (S.C.Y. Lu and R. Komanduri Eds.), *The Winter Annual Meeting ASME*, Anaheim, CA, pp. 275–286.

IPAD, 1977a, "Integrated Information Processing Requirements (D6–IPAD–70040–D)", Development of Integrated Programs for Aerospace–Vehicle Design (IPAD), NASA, June.

IPAD, 1977b, "IPAD Requirements (D6–IPAD–70040–D)", Development of Integrated Programs for Aerospace–Vehicle Design (IPAD), NASA, December.

Ishii, K., 1992, "The Fourth Annual Life–cycle Design Sponsors Meeting", June 9, 1992, Dept. of Mechanical Engineering, The Ohio State University.

Ishii K., and Miller R.A., 1992, "Design Representation for Manufacturability Evaluation in CAD: Beyond Feature–Based Design", *Computer in Engineering*, Volume 1, ASME 1992, pp. 37–44.

Ishii, K. and S. Mukherjee, 1992, "Computer–aided Life–cycle Design Compatibility Analysis", *Proceedings of the NSF Design and Manufacturing Conference*, SME, January, 1992 , Atlanta, GA.

ISO 1986, The Ottawa Report on Reference Models for Manufacturing Standards, ISO, TC184/SC5/WG1 N51.

ISO 9000, NISTIR4721 – Questions and Answers on Quality, the ISO 9000 Standard Series, Quality System Registration, and Related Issues, NISTIR 5122 – More Questions and Answers on the ISO 9000 Standard Series and Related Issues, U.S. Department of Commerce, *National Insititure of Standards and Technology*, April 1993.

ISO CD 10303 – 1, 1993, *Product Data Representation and Exchange – Part 1: Overview and Fundamental Principles.*

ISO DIS 10303–11, 1993, *EXPRESS Language Reference Manual.*

ISO/IEC 10746–1 I ITU–T Rec. X.901, "Draft Recommendation X.901: Basic Reference Model of Open Distributed Processing – Part 1: Overview and guide to use", ISO/IEC JTC1/SC21/WG7 N 755, 1993.

ISO/IEC 10746–2 I ITU–T Rec.X.902, "Draft Recommendation X.902: Information Technology – Basic Reference Model of Open Distributed Processing Part 2: Descriptive Model", ISO/IEC JTC1/SC21/WG7, 1993.

ISO/IEC 10746–3 I ITU–T Rec.X.903, "Draft Recommendation X.903: Basic Reference Model of Open Distributed Processing – Part 3: Prescriptive Model", ISO/IEC JTC1/SC21/WG7/N8125, 1993.

ISO TC184 SC4 WG8/N9, "KCIM: Resource Model", Presentation by B.R. Katzy, Seattle, 8–17 April, 1992.

ISO TC184 SC4 WG8/N13, Project 2: "Resource Usage Management", Draft Scope presented at the London Meeting by B.R. Katzy, June, 1992.

Iwata K., Kakino Y., Oba F. and Sugimura N., 1980, "Development of non–part family type computer aided production planning system CIMS/PRO", *Advanced manufacturing technology*, (P. Blake Ed.), North Holland.

Jakubowski R., 1982, "Synthactic characterization of machine parts", *Cybernetics and Systems: An International Journal*, 13, pp. 1–24.

Johnson R.H., 1987, *Commercial Product Data Management Systems: A Survey of Needs and Systems*, CIMdata Inc., Ann Arbor, Michigan.

Jorgenson B. R., 1992, "Model Repository Technology for Model Integration, Enterprise Integration Modeling", *Proceedings of the First International Conference*, The MIT Press, October.

Jorysz H.R., and Vernadat F.B., 1990a, "CIM–OSA Part 1: total enterprise modelling and function view", *Int. J. Computer Integrated Manufacturing*, Vol. 3, Nos. 3–4, pp. 144–156.

Jorysz H.R., and Vernadat F.B., 1990b, "CIM–OSA Part 2: information view", *Int. J. Computer Integrated Manufacturing*, Vol. 3, Nos. 3–4, pp. 157–167.

Joshi S., Vissa N.N., and Chang T.C., 1989, "Expert process planning system with solid model interface", *Knowledge–based Systems in Manufacturing*, A. Kusiak (Ed.), Taylor and Francis, pp. 111–136.

Jovane F., Alting L., Armillotta A., Eversheim W., Feldman K., Seliger G., and Roth N., 1993, "A Key Issue in Product Life Cycle: Disassembly", *Annals of the CIRP* Vol. 42/2/1993, pp. 651–658.

Judson D.L., 1986, "Integrated Information Support Systems", Integrated Information Support System (IISS): An Evolutionary Approach to Integration, Manufacturing Technology Division, Materials Laboratory, Air Force Wright Aeronautical Laboratories, 1985.

Kahaner D., and Lu S., 1993, "First CIRP International Workshop on Concurrent Engineering for Product Realization", *Concurrent Engineering Research in Review*, Volume 5, pp. 6–14.

Kalpakjian, S., Manufacturing processes for engineering materials, Addison–Wesley, 1984.

Karandikar H.M., Wood R.T., and Byrd J., 1992, "Process and Technology Readiness Assessment for Implementing Concurrent Engineering", CERC Technical Report Series, *CERC–TR–TM–92–004*.

Keegan, D.P., Eiler, R.G., and Jones, C.R., 1989, "Are Your Performance Measures Obsolete", *Management Accounting*, June, pp. 45–50.

Kerr, R.M., and Greenhalgh G.R., "Aspects of manufacturing strategy", *Production Planning & Control*, 1991, Vol. 2, No. 3, 194–206.

Kidd P., 1992, *Organization, people and technology in European manufacturing, Commission of the European Communities*, Report EUR 13967 EN.

Kidd P.T., 1994a, "Agile Manufacturing: Key Issues", *in Advances in Agile Manufacturing*, P.T. Kidd and W. Karwowski (Eds.), IOS Press, pp. 29–32.

Kidd P.T., 1994b, *Agile Manufacturing: Forging New Frontiers*, Addison–Wesley.

Kief H.B., and Waters T.F., 1992, *Computer Numerical Control*, Glencoe, Macmillan/McGraw–Hill.

Kimura F., 1991, "Software for Product Realization", *International Symposium for International Trends in Manufacturing Towards the 21st Century*, Oct. 18, Berlin, Germany.

Kimura F., 1992, "Future Perspective of CAD/CAM Research and Development in Japan", CAD'92, Neue Konzepte zur Realisierung anwendungsorientierter CAD–Systeme, *GI Informatik aktuell*, Springer–Verlag, Berlin–Heidelberg–New York.

Kimura F., 1993a, "A Computer–Supported Framework for Concurrent Engineering - An Approach based on Virtual Manufacturing – ", Information Infrastructure Systems for Manufacturing, *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 345–359.

Kimura F., 1993b, "Product and Process Modelling as a Kernel for Virtual Manufacturing Environment", *Annals of the CIRP* , Vol. 42/1/93, pp. 147–150.

Kimura F., and Suzuki H., 1986, "Variational product design by constraint propagation and satisfaction in product modelling", *Annals of the CIRP*, Vol. 35/1, pp. 75–78.

Kimura F., and Suzuki H., 1989, "A CAD System for Efficient Product Design Based on Design Intent", *Annals of the CIRP*, Vol. 38/1, pp. 149–152.

Kimura F., Suzuki H., Ando H., T. Sato, and Kinosada A., 1987 "Variational geometry based on logical constraints an its applications to Product Modelling", *Annals of the CIRP*, Vol. 36/1, pp. 65–68.

Kinstrey M., Lee V.K., Lewis J.W., Salant E.L., Sarachan B.D., and Wilson P.R., 1990, "PPO Management", *Proceedings 2nd National Simposium on Concurrent Engineering*, February 7–9, Morgantown, West Virginia, pp. 67–90.

Klein J., and Scheer A.D., 1990, "Information modelling – the basis for integrated information systems", *Workshop Implementing CIM*, 3–4 December, Saabrucken, Germany.

Klittich M., 1990, "CIMOSA Part 3: CIMOSA Integrating Infrastructure – The Operational Basis for Integrated Manufacturing Systems", *Int J. Computer Integrated Manufacturing*, Vol. 3, Nos. 3–4, pp. 168–180.

Kosanke K., 1992, "CIMOSA – A European Development for Enterprise Integration Part 1: An Overview", in Enterprise Integration Modelling: *Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 179–188.

Krause F–L., and Major F., 1988, "Intelligent Design Support Environment: An Integrating Concept of Methods and Models", *Preprints of the second IFIP WG 5.2 Workshop on Intelligent CAD*, September 19–22th, 1988, Cambridge, UK.

Krause F–L., Jansen H., Bienert M., and Major F., 1989, "System Architectures for Flexible Integration of Product Gestaltung", *Preprints of the International GI–IFIP Symposium of Advanced Geometric Modelling for Engineering Applications*, Production Technology Centre Berlin, November 8–10, Berlin (West), FRG, pp. 407–425.

Krause F–L., Kimura F., Kjellberg T., and Lu S.C.–Y., 1993, "Product Modelling", Keynote papers, *Annals of the CIRP*, Keynote paper, Vol. 42/2/1993, pp. 695–706.

Krause F–L, Ulbrich A., and Vosgerau F.H., 1991, "Feature–Based Approach for the Integration of Design and Process Planning Systems", *Product Modelling for Computer Aided Design and Manufacturing*, North–Holland, pp. 285–299.

Krause F–L., Vosgerau F.H., and Yaramanoglou N., 1987, "Using technical Rules and Features in Product Modeling", IFIP WG 5.2, *First International Workshop on Intelligent CAD*, Cambridge, MA, October 6–8.

Kulweic, R.A. (ed.), 1984, *Material Handling Handbook*, New York: John Wiley & Sons.

Kung H., 1984, "An investigation into development of process plans from solid geometric modeling representation", Ph.D. Thesis, Oklahoma State University.

Kyprianou L.K., 1983, "Shape classification in computer–aided design", Ph.D. Thesis, University of Cambridge, Cambridge, UK.

Lakko T., and Mantyla M., 1993, "Feature modelling by incremental feature recognition", *Computer–Aided Design*, Volume 25, Number 8, August, pp. 479–492.

Lawrence A., 1990, "Engineering Database Management", *Industrial Computing*, May, pp. 22–24.

Lee M.H., 1993, "The knowledge–based factory", *Artificial Intelligence in Engineering*, 8, pp. 109–125.

Lee Q., 1992,"Manufacturing focus – comprehensive view", *In Manufacturing Strategy – Process and content*, Chapter 15, C.A. Voss (ed.), Chapman & Hall, pp. 297–317.

Lee R.J.V., 1993, "Design for manufacture of injection moulded products", PhD Report, Manufacturing Engineering Department, Loughborough University of Technology.

Leech M.J., Weston R.H., Hodgson A., Clements P., and Ryan A., 1991, "Incremental building of CIM Systems", *International Conference on CIM*, Singapore, September.

Lee–Mortimer A., 1991, "Simultaneous engineering at NCR", *TQM Magazine*, April, 3(2), pp. 99–104.

Li R.K., 1988, "A part–feature recognition system for rotational parts", *International Journal of Production Research*, Vol. 26, No. 9, pp. 1451–1475.

Lienhart D., 1990, "Frameworks and the CAD Framework Initiative", in Engineering Data Management: The Technology for Integation, C.A. Bon, W.S. Rasdorf, R.E. Fulton (Eds.), *ASME CE Conference*, August 5–9, pp. 11–13.

Linington P.F., 1992, "Introduction to the Open Distributed Processing Basic Reference Model", *Proceedings of the 1st Int. Workshop ODP*, University of Kent, Vol. 1, pp. 3–13.

Linton, L.R., Hall, D., Hutchison, K., Hoffman, D., Evanczuk, S. and Sullivan P., 1992, "First Principles of Concurrent Engineering: A Competitive Strategy for

Electronic Product Development", CALS/CE Electronic Task Group, May 1992, *CALS Technical Report 005*.

Liu C.R., and Srinivasan, 1984, "Generative process planning using syntactic pattern recognition", *Computers in Mechanical Engineering*, March, pp. 63–66.

Lu S.C–Y., 1992, "Research, Development and Implementation of Knowledge processing Tools to Support Concurrent Engineering Tasks". In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 3–25.

Luby S.C., Dixon J.R., Simmons M.K., 1986, "Designing with Features: Creating and Using a Features Data Base for Evaluation of Manufacturability of Castings", *Proceedings 1986 ASME International Computer in Engineering Conference*, Chicago, Illinois, pp. 285–292.

Lynch M., 1992, *Computer Numerical Control for Machining*, Mc–Graw–Hill Inc., New York.

Mantyla M., 1988, *An Introduction to Solid Modeling*, Computer Science Press, Rockville, Inc.

Mantyla M., 1989, "Directions for Research in Product Modelling", *Computer Applications in Production and Engineering*, F. Kimura and A. Rolstadas (Editors), Elsevier Science Publishers B.V. (North–Holland).

Mantyla M., 1993, "Towards Open Architecture Concurrent Engineering Frameworks", *Otaniemi 1993/TKO–B96*, Department of Computer Science, Faculty of Information Technology, Helsinki University of Technology.

Material Handling Engineering Reference Guide, *Plant Engineering*, March, F.1 1–20.

Matson, J.O., Mellichamp, J.M. and Swaminathan, S.R., 1992, "EXCITE: Expert consultant for in–plant transportation equipment", *International Journal of Production Research*, Vol. 30, No. 8, 1969–1983.

Maul, R., Hughes D., Childe S., Weston N., Tranfield D., and Smith S., 1990, "A Methodology for the Design and Implementation of Resilient CAPM Systems", *International Journal of Operations & Production Management*, Vol 10, No. 19, pp. 27–36.

Mayer R.J., and Painter M.K., 1991, "Roadmap for Enterprise Integration", Conference *Proceedings AUTOFACT' 91*, November 10–14, 1991, Chicago, Illinois, SME, pp. 7:1–7:26.

Mayer R.L., Menzel C.P., and Mayer P.S.D., 1990, "IDEF3 Technical Report", *Knowledge Based Systems Lab*, Texas A&M.

McKay A., and Bloor M.S., 1991, "The role of product models in efective CADCAM", Proceedings of the Institution of Mechanical Engineers, *Effective CADCAM'91*, European Conference, 6–7 November, Coventry, UK, pp. 113–119.

McMahon C., and Browne J., 1993, *CADCAM: From principles to practice,* Addison–Wesley.

Menzel C., Mayer R.J., and Sanders L.K., 1992, "Representation, Information Flow, and Model Integration", *in Enterprise Integration Modelling: Proceedings of the First International Conference,* C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 131–141.

Mertins K., Sussenguth W., and Jochem R., 1992, "An Object Oriented Method for Integrated Enterprise Modeling as a Basis for Enterprise Coordination", *in Enterprise Integration Modelling: Proceedings of the First International Conference,* C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 249–258.

META Software Corporation, 1990, *Design/IDEF User's Manual,* USA

Miles B., 1990, "Design for manufacture techniques help the team make early decisions", *Journal of Engineering Design,* 1 (4) pp. 365–371.

Molina A., Mezgar I., and Kovacs G., 1992, "Object Knowledge Representation Models for Concurrent Design of FMS", *in Human Aspects in Computer Integrated Manufacturing,* G.J. Olling, F. Kimura (Editors), Elsevier Science Publishers B.V. (North Holland), 1992 IFIP, pp. 779–788.

Molina A., 1993, "Relevant British Standards Review", Turin, Italy, February 22–26, *ISO TC184 SC4 WG8/N26.*

Molina A., A.H. Al–Ashaab, Ellis T.I.A, Young R.I.M, Bell R., 1994, "A Review of Computer Aided Simultaneous Engineering Systems", Accepted for publication in the *Journal of Research in Engineering Design,* June 1994.

Moore, H.R., and Kibbey, D.R., 1982, *Manufacturing Materials and Processes,* Third Edition, Grid Publishing Inc..

Morgan E., 1989, "Through MAP/TOP to CIM", Department of Trade and Industry's publication , Moore & Matthes (Printers) Ltd.

moses–core–mm–3, T.I.A. Ellis, and A. Molina, "Booch Representation of the Manufacturing Model", MOSES Project Core Document – Manufacturing Model ~ moses–core–mm–3, MOSES Project, November 1994.

moses–core–mm–4, T.I.A. Ellis, and A. Molina, "Express and Express–G Representation of the Manufacturing Model", MOSES Project Core Document – Manufacturing Model ~ moses–core–mm–4, MOSES Project, November 1994.

Murakami T., and Nakajima N., 1987, "Research on Design–Diagnosis Using Feature Description", IFIP WG 5.2, *First International Workshop on Intelligent CAD*, Cambridge, MA, October 6–8.

Murata T., 1989, "Petri–Nets: Properties, Analysis and Applications", *Proceedings of the IEEE*, Vol. 77, No 4, April, pp. 541–580.

Muthsam H., and Mayer C., 1990, "An expert system for process planning of prismatic workpieces", *Proc. 1st Conf. Artificial Intelligence and Expert Systems in Manufacturing*, March, pp. 211–220.

Naik, B., and Chakravarty A.K., 1992, "Strategic acquisition of new manufacturing technology: a review and research framework", *International Journal of Production Research*, Vol. 30, No. 7, 1575–1601.

Nevins J.L. and Whitney D.E., 1989, *Concurrent Design of Product and Processes: A Strategy for the Next Generation in Manufacturing*, McGraw–Hill Publishing Company, N.Y.

Nichols, K., 1991, "Simultaneous engineering at DEC", *TQM Magazine*, April, 3(2), pp. 105–110.

Nnaji B.O., Liu H–C, and Rembold U., 1993, "A product modeller for discrete components", *Int. J. Prod. Res.*, Vol. 31, No. 9, 2017–2044.

Nof S.Y., 1994, "Critiquing the potential of object orientation in manufacturing", *Int. J. Computer Integrated Manufacturing*, Vol. 7, No. 1, 3–16.

Objectivity/DB, 1991, *Reference Manuals*, Objectivity Inc.

O'Grady P., Young R.E., Greef A., and Smith L., 1991, "An advice system for concurrent engineering", *International Jounal of Computer Integrated Manufacturing*, Vol. 4, No. 2, pp. 63–70.

Oh J.H., and Lee K., 1990, "Automatic Dimensioning from 3D Solid Model with Feature Extraction", *Proceedings of 1990 ASME Design Technical Conferences –*

16th Design Automation Conference, Chicago, Ill, September 16–19, DE–Vol. 23–1, pp. 115–119.

Onosato M., and Iwata K., 1992, "VirtualWorks: Building a Virtual Factory with 3–D Modelling and Object Oriented Programming Techniques", *Proceedings IFAC–INCOM'92*, pp. 281–286.

Onosato M., and Iwata K., 1993, "Development of a Virtual Manufacturing System by Integrating Product Models and Factory Models", *Annals of the CIRP*, Vol. 42/1/1993, pp. 475–478.

Onosato M., Iwata K., and Nishida S., 1992, "Modelling Time Information in Manufacturing", *JSME International Journal*, Series III, 35–2, pp. 207–213.

Orr K., anf Gane C., 1989, "Edward Yourdon, Peter P. Chen, Larry L. Constantine, Methodology: The Experts Speak", *BYTE*, April, pp. 221–233.

Owen J., and Brett S., 1992, "Introduction to STEP", *CADDETC Seminar*, Leeds, England, July.

Painter M.K., Mayer R.J., and Cullinane T.P., 1991, "The Many Faces of Concurrent Engineering", *AUTOFACT'91*, Conference Proceedings, November 10–14, 1991, Chicago, Illinois, SME, pp. 17:1–17:25.

Painter, P (Ed.), 1990, CAD Framework Initiative, Architecture Technical Subcommittee: Requirements Task Group, "CAD Framework Users, Goals, and Objectives", Version 0.8, Draft Proposal Document, May 14.

Pande S.S., and Walverkar M.G., 1990, "PRICAPP: A computer assisted process planning system for prismatic components", *Int. J. Prod. Res.*, Vol. 28, No. 2, pp. 279–292.

Park M.W., Vosniakos G.C., and Davies B.J., 1990, "Automatic Feature Extraction in Wire–Frame Models of 2 1/2–D Prismatic Parts", *Proceedings 28th International Matador Conference*, 18–19 April, Manchester, England, pp. 83–89.

Peterson J.L., 1981, *Petri Nets Theory and the Modelling of Systems*, Prentice–Hall.

Petrie C., 1992, "Appendix A ICEIMT Workshop Calls for Participation", *in Enterprise Integration Modelling: Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 545–560.

Potts, J.S., 1990, "Choosing the right quality programme", *Quality Progress*, October, 23 (10), pp. 62–64.

Pratt M.J., 1988, "Synthesis of an Optimal Approach to Form Feature Modeling", *Proceedings of ASME International Computers In Engineering Conference*, San Francisco, CA, August, pp. 263–274.

Pratt M.J., and Wilson P.H., 1985, "Requirements for Support of Form Features in A Solid Modeling System", *CAM–I Report R–85–ASPP–01*, Arlington, Texas.

Querenet B., 1992, "CIMOSA – A European Development for Enterprise Integration Part III: Enterprise Integrating Infrastructure", *in Enterprise Integration Modelling: Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 205–215.

Ranky, P., *The Design and Operation of FMS – Flexible Manufacturing Systems*, IFS (Publications) Ltd., UK, North Holland Publishing Company, *1983.*

Rational ROSE, 1993, *Object–Oriented Software Engineering*, Version 1.1, Rational.

Reddy R.Y.V., Srinivas K., Jagannathan V., and Karinthi R., 1993, "Computer Support for Concurrent Engineering", *IEEE Computer*, 26, 1, pp. 12–16.

Regnier P., Fenie P., and Doumeingts G., 1993, "A methodology to design a Reactive Production Management System", *Presented at the IEEE System, Man and Cybernetics Conference.*

Rehg J., Elfes A., Talukdar S., Woodbury R., Eisenberger M., and Edahl R., 1988, "CASE: Computer–aided simultaneous engineering", *Artificial Intelligence in Engineering: Design*, (Ed.) J.S. Gero, Elsevier, pp. 339–360.

Rembold U., Blume C., and Dillman R., 1985, *Computer–integrated manufacturing technology and systems*, M. Dekker Publisher.

Rembold U., and Nnaji B., 1991, "The role of manufacturing model for the information technology of the factory of the 1990s", *Journal of Design and Manufacturing*, (1991), 1, 67–87.

Rembold U., Lueth T., and Ogasawara T., 1994, "Intelligent Manufacturing Components for Future CIM Systems", Survey Paper, Preprints, *2nd IFAC/IFIP/IFORS Workshop Intelligent Manufacturing Systems – IMS'94*, (Ed. P. Kopacek), Vienna – Austria – June 13–15, 1994, pp. 1–10.

Remmerswaal J.L., 1990, "Report on the Round Table held in Berlin at the 40th General Assembly, 1990", *Annals of the CIRP*, Vol. 39/2/1990, pp. 589–593.

Requicha A.A.G., 1980, "Representations for rigid solids: theory, methods, and systems", *ACM Computing Surveys*, 12, 4, pp. 437–464.

Requicha A.G.G., and Voelcker H.B., 1982, "Solid Modeling: a historical summary and contemporary assessment", *IEEE Computer Graphics and Applications*, March, pp. 9–24.

Rumbaugh J., Blaha M., Premerla W., Eddy F., and Lorensen W., 1991, *Object–oriented modeling and design*, Prentice Hall Inc.

Sarkis J., and Lin L, 1994, "An IDEFo functional planning model for the strategic implementation of CIM systems", *Int. J. Computer Integrated Manufacturing*, Vol. 7, No. 2, 100–115.

Sata T., Kimura F., Suzuki H., and Fujita T., 1985, "Designing Machine Assembly Structure Using Geometric Constraints in Product Modelling", *Annals of the CIRP*, Vol. 34/1, pp. 169–172.

Scheer A.W., 1989, *Enterprise–Wide Data Modelling: Information Systems in Industry*, Springer–Verlag Berlin Heidelberg.

Scheer A.–W., 1993, *Architecture of Integrated Information Systems – Foundations of Enterprise Modelling*, Springe–Verlag, Berlin.

Schenck D., and Wilson P., 1994, *Information Modeling the EXPRESS Way*, Oxford University Press.

Schey, J. A., 1987, *Introduction to manufacturing processes*, Second Edition, McGraw–Hill.

Schulte R., Padmanabham S., and Devgun M.S., 1992, "A feature–driven, process–based approach to the integration of CAD/CAM in wireframe models", *Int. J. Prod. Res.*, Vol. 30, No. 5, pp. 1005–1028.

Shah J.J., and Mathew A., 1991, "Experimental investigation of the STEP Form–Feature Information Model", *Computer Aided Design*, Vol. 23, No. 4, May, pp. 282–296.

Shah J.J., and Rogers M.T., 1988, "Functional requirements and conceptual design of feature–based modeling system", *Computer–Aided Engineering Journal*, February, pp. 9–15.

Shah J.J., and Wilson P.R., 1988, "Analysis of knowledge abstraction, representation and interaction requirements for computer aided engineering", *Proceedings of*

ASME International Computers in Engineering Conference, San Francisco, CA, August, pp. 17–24.

Shah J.J., Sreevalsan P., Rogers M., Billo R., and Mathew A., 1989, "Current Status of Features Technology", *CAM–I Report R–88–GM–04–1*, Arlington, Texas, November.

Shaw N.K., Bloor M.S., and de Pennington A., 1989, "Product Data Models", *Research in Engineering Design*, June/July, pp. 43–50.

Sheridan J.H., 1993, "Agile Manufacturing", in Agile Manufacturing: stepping beyond lean production, *Industry Week*, Vol 342, No 8 April 19, pp. 30–46.

Siegal B., 1991, "Organising for a successful CE process", *Industrial Engineering*, December, 23(12) pp. 15–19.

Skinner, W., 1974, "The Focused Factory", Harvard *Business* Review", Vol. 51, No. 3, pp. 113–21.

Smith H.C., 1985, "Database Design: Composing Fully Normalized Tables from a Rigorous Dependency Diagram", *Communications of the ACM*, Vol. 28, No. 8, August.

Smith J.M., and Smith D.C.P., 1977, "Database Abstractions: Aggregations and Generalization", *ACM Trans. on Database Systems*, Vol. 2., No. 2, pp. 105–133.

So T., Jagannathan V., and Raman R.S., 1992, "The Role of Configuration Management Systems in a Concurrent Engineering Environment", CERC technical Report Series, *CERC–TR–TM–92–015*.

Sohlenius G., 1992, "Concurrent Engineering", Keynote Paper, *Annals of the CIRP* Vol. 41/2/1992.

Sowa J.F., 1984, *Conceptual Structures: Information Processing in Mind and Machine*, Addison–Wesley Publishing Company.

Spur G., Krause F.L., and Armbrust P., 1986, "Product Models as Basis for Integrated Design and Manufacturing", *Int. J. Mach. Tool Des. Res.*, Vol. 26, No. 2, pp. 171–178.

Spur G., Krause F.L., and Lehmann C.M., 1989a, "Integration of methods for knowledge processing and geometric modelling", *Proceedings of the Institution of Mechanical Engineers*, Fourth European Conference, Effective CADCAM, 6–7 December, London, pp. 57–63.

Spur G., Krause F.L., Mayr R., Müller G., and Schliep W., 1979, "A Survey about Geometric Modelling Systems", Keynote Paper, *Annals of the CIRP,* Vol 28/2, pp. 519–538.

Spur G., Mertins K., and Sussenguth W., 1989b, "Integrated Information Modelling for CIM", *CIM–Management 2/89*, Munchen, Oldenbourg, pp. 36–42.

Srihari K., and Greene T.J., 1988, "Computer–Aided Process Planning – A CAD/CAM Interface", *CIM Review*, Summer, pp. 47–54.

Srinivasan R., Liu C.R., and Fu K.S., 1985, "Extraction of manufacturing details from geometric models", *Computers In Industrial Engineering*, 9, 2, pp. 125–133.

Staley S.M., Henderson M.R., and Anderson D.C., 1983, "Using syntactic pattern recognition to extract features from a solid geometric data base", *Computers in Mechanical Engineering*, September, pp. 61–66.

STEP Tool Kit, 1993, STEP Tools, Inc., Rensselaer Technology Park, NY.

Steudel, H.J., 1984, "Computer Aided Process Planning: Past, Present, and Future", *International Journal of Production Research*, 22 (2), pp. 253–266.

Stevenson I., 1992, "IT Based Tools for Concurrent Engineering", *Proceedings of the Eighth CIM–Europe Annual Conference*, 27–29 May, 1992, Birmingham, UK, pp. 157–166.

Stoll H.W., 1988, "Design for Manufacture", *Manufacturing Engineering*, January, pp. 67–73.

Stroustrup B., 1986, *The C++ Programming Language*, Second Edition, Addison–Wesley.

Tavakoli, A., 1993, "Feature Based Workshop Oriented NC Planning for Asymmetric Rotational Parts", PhD Thesis, Manufacturing Engineering Department, Loughborough University of Technology.

Taylor S.J., 1990, "3D CADCAM in Manufacturing", CADCAM'90 Conference Proceedings, 27–29 March, Birmingham, pp. 428–439.

Teory T.J., et al., 1986, "A Logical Design Methodology for Relational Database Using the Extended Entity–Relationship Model", *Computing Survey*, Vol. 18, No. 2, June.

Tipnis V.A., 1993, "Evolving Issues in Product Life Cycle Design", *Annals of the CIRP,* Vol. 42/1/1993, pp. 169–173.

Tompkins, J.A., and White, J.A., 1984, *Facilities Planning*, New York: John Wiley & Sons.

Tsukune H., Tsukamoto M., Matshisita T., Tomita F., Okada K., Ogasawara T., Takase K., Yuba T., 1993, "Modular Manufacturing", *Journal of Intelligent Manufacturing*, 4, pp. 163–181.

Tucker D.E., and Leonard R., 1994, "Overcoming the Cultural Barriers to Implementing Concurrent Engineering", *Advances in Agile Manufacturing*, P.T. Kidd nad W. Karwowski (Eds.), IOS Press, 1994, pp. 67–70.

Unger M.B., and Ray S.R., 1988, 'Feature–based process planning in the AMRF", *Proceedings of the ASME International Computers in Engineering Conference*, San Francisco, CA, August, pp. 563–569.

Vallespir B., Merle C., and Doumeingts G., 1993, "GIM: A Technico–Economic Methodology to Design Manufacturing Systems", *Control Eng. Practice*, Vol 1., No. 6, pp. 1031–1038.

Van Griethuysen J.J., 1992, "Enteprise modelling, a necessary basis for modern information systems", *Open Distributed Processing*, J. de Meer, V. Heymer and R. Roth (Editors), Elvesier Science Publishers B.V. (North–Holland), IFIP., pp.

Van Houten F.J.A.M., 1992, "Manufacturing Interfaces", Keynote Paper, *Annals of the CIRP*, Vol. 41/2/1992.

Van Houten F.J.A.M., van 't Erve A.H., and Kals H.J.J., 1989, "PART a Feature Based CAPP System", SPIN Project, Report PA89–008, February.

Verheijen, G. and VanBekkum, J., 1982, "NIAM: an information analysis method", in "Information Systems Design Methodology", edited by Olle, Sol and Verrijn–Stuart, North Holland.

Vernadat F.B., 1992, "CIMOSA – A European Development for Enterprise Integration Part 2: Enterprise Modelling", *in Enterprise Integration Modelling: Proceedings of the First International Conference*, C.J. Petrie (ed.), The MIT Press, Cambridge, Massachusetts, pp. 189–204.

Voss C.A., Russell V., and Twigg D., 1991, "Implementation issues in simultaneous engineering", *International Journal of Technology Management*, 6 (3/4), pp. 293–302.

Vujosevic R., and Kusiak A., 1991, "Data Base Requirements for Concurrent Design Systems", In Engineering Databases: An Engineering Resource, *Proceedings of the*

ASME International Computer in Engineering Conference & Exposition, (ed.) V. Saxena, August 18–22, Santa Clara, California, pp. 109–113.

Walker H., and West R., 1990, "Integrating CAD and CAM Through the Feature Based Methodology for Design and Manufacture", *Proc. 28th Int. Matador. Conf*, 18–19 April, Manchester, England, pp. 273–281.

Wang H–P., and Wysk R.A., 1988, "AIMSI: A Prelude to a New Generation of Integrated CAD/CAM Systems", *International Journal of Production Research* , Vol. 26, No. 2, pp. 119–131.

Warnecke H.J., 1993, *The Fractal Company: A Revolution in Culture*, Springer–Verlag.

Weck M., Eversheim W., Konig W., and Pfeifer T., *Production Engineering – The Competitive Edge*, Butterworth–Heine mann, 1991.

Weston R.H., 1993, "Steps towards enterprise–wide integration: a definition of need and first–generation open solutions", *Int. J. Prod. Res.*, Vol. 31, No. 9, pp. 2235–2254.

Wheeler R., Burnett R.W., et al., 1991, "Concurrent engineering: success stories in instrumentation, communications", IEEE Spectrum, July 1991, 28(7), pp. 32–37.

Wilhelm R., and Lu S.C.Y., 1992, "Tolerance Synthesis to Support Concurrent Engineering", *In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory*, University of Illinois at Urbana–Champaign, pp. 47–54.

Williams T.J., 1991, *The Purdue Enterprise Reference Architecture*, Report Number 154, Purdue Laboratory for Applied Industrial Control, December.

Williams T.J., 1993, "The Purdue Enterprise Reference Architecture", Information Infrastructure Systems for Manufacturing, *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 43–62.

Williams T.J., Bernus P., Brosvic J., Chen D., Doumeingts G., Nemes L., Nevins J.L., Vallespir B., Vliestra J., and Zoetekouw D., 1993, "Architectures for Integrating Manufacturing Activities and Enterprises", Information Infrastructure Systems for Manufacturing, *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 3–16.

Wilson P., 1990, *Computer Supported Cooperative Work: An Introduction*, P.A. & CCTA Advanced Concepts Branch, Intellect Publishers, Oxford, UK.

Wilson P., 1993, "Processing Tools for EXPRESS", available electronically from the SOLIS system maintained at NIST.

Winner R.I., Pennell J.P., Bertrand H.E., and Slusarczuk M.M.G., 1988, "The Role of Concurrent Engineering in Weapon System Acquisition", *IDA-Report R-338.*

Wirfs-Brock R., Wilkerson B., and Wiener L., 1990, *Desigining object-oriented software*, Prentice-Hall.

Woo, T.C., 1975, Computer understanding of designs, Ph.D. Thesis, University of Illinois, Urbana-Champaign.

Woo, T.C., 1977, "Computer aided recognition of volumetric design", *Advances in Computer Aided Manufacture*, (Ed.) D. McPherson, Amsterdam, North Holland.

Woo, T.C., 1982, "Feature extraction by volume decomposition", *Conference on CAD/CAM Technology in Mechanical Engineering*, MIT, MA, March.

Woo T.C., 1984, "Interfacing solid modeling to CAD and CAM: data structures and algorithms for decomposing a solid", *IEEE Computer*, December, pp. 44-49.

Woodgate S., 1991, "Concurrent Engineering: it works, and here is the proof", *Engineering Computers*, 10 (1), pp. 34, 35, 37.

Woodwark J., 1988, "Shape models in computer integrated manufactured – a review", *Computer Aided Design*, June, pp. 103–112.

Wu, J.K., T.H. Liu, and G.W. Fisher, 1992, "An Integrated PDES/STEP Based Information Model for CAE and CAM Applications", *Technical Report R-129*, Center for Simulation and Design Optimization of Mechanical Systems, The University of Iowa.

Yang Y., 1991, "Three-Schema Implementation in STEP: Standard for the Exchange of Product Model Data", In Engineering Databases: An Engineering Resource, *Proceedings of the ASME International Computer in Engineering Conference & Exposition*, (ed.) V. Saxena, August 18-22, Santa Clara, California, pp. 1-5.

Yoshikawa H., 1994, "Intelligent Manufacturing Systems: Technical Cooperation That Transcend Cultural Differences", *Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing*, DIISM'93, Tokyo, Japan, 8-10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North-Holland, pp. 19-38.

Young, R.I.M. and Bell R., 1992, Machine Operation Planning in Product Modelling Environment, *International Journal of Production Research*, Vol. 30, No. 11, pp. 2487–2513.

Young, R.I.M., "Manufacturing Models", *In Information Systems for Competitive Manufacture*, EPSRC Seminar, September 1994.

Yourdon, E., and Constantine L.L., 1979, *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design* – Edition 2, Prentice–Hall, Englewood Cliffs, NJ.

Zachman J.A., 1987, "A framework for information systems architecture", *IBM Systems Journal*, vol. 26, no. 3, 1987, pp. 276–292.

Zust R., and Wagner R., 1992, "Approach to the Identification and Quantification of Environmental Effects during Product Life", *Annals of the CIRP*, Vol. 41/1/1992, pp. 473–476.

APPENDICES

Appendix A

A Review of Computer Aided Simultaneous Engineering Systems

This appendix presents the review paper entitled "A Review of Computer Aided Simultaneous Engineering Systems" wrote by the author in collaborations with LUT members of the MOSES Research Project. This paper has been accepted for publication in *Research in Engineering Design Journal* by Springer–Verlag.

A Review of Computer Aided Simultaneous Engineering Systems

A. Molina, A. H. Al–Ashaab, T.I.A. Ellis, R.I.M. Young, R. Bell
Manufacturing Engineering Department, Loughborough University of Technology, Loughborough, Leicestershire, LE11 3TU, U.K.

Abstract. World–wide markets are becoming increasingly competitive and in order to sustain market share organisations are developing a customer oriented approach for designing and producing high quality, high value products. The philosophy of Simultaneous Engineering has been proposed as a potential means of improving product development practice. This philosophy involves simultaneously satisfying the functionality, reliability, produceability, and marketability concerns of new products in order to reduce product development time and cost, and to achieve higher product quality and value. In this paper the concept, objectives and principles of Simultaneous Engineering are introduced. The research that has, and is, being undertaking into computer aided systems for the support of Simultaneous Engineering is presented and reviewed.

Keywords: concurrent engineering, information technology, information modelling, integration, framework, simultaneous engineering, system architecture

1. The Concept, Objectives and Principles of Simultaneous Engineering

Increasing competition in the world market place has forced companies to look for new means of improving the quality of products, decreasing production costs and reducing the time taken to introduce new products. Simultaneous Engineering (SE) or Concurrent Engineering (CE) seems to be the key to achieving and sustaining a competitive advantage through the development of high quality, highly functional products that are produced efficiently through the synergy of integrated product and process design, whilst also considering multiple life cycle factors such as functionally, serviceability, manufacturability, marketability and recyclability.

In this philosophy the objectives focus on (Winner et al. 1988, Nevins and Whitney 1989):

- improvement of quality
- reduction of life–cycle costs
- reduction of development lead times

Simultaneous Engineering has been defined by Winner et al. (1988) as "a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements".

Cleetus (1992) has proposed a new definition: "CE is a systematic approach to integrated product development, that emphasizes response to customer expectations and embodies team values of cooperation, trust, and sharing in such a manner that decision making proceeds with large intervals of parallel working by all life–cycle perspectives, synchronized by comparatively brief exchanges to produce consensus".

In both definitions the importance of addressing the different aspects of SE in a systematic manner is emphasized. In order to be systematic, Simultaneous Engineering should be characterized by applying principles which introduce cultural, human and organizational changes within the enterprise through the use of formal methodologies, in some cases, supported by information technology. The SE principles which guide the enterprise changes have been classified as follows (Painter et al. 1991, Linton et al. 1991):

- Organization principles: new organizational structures, customer focus attitude, discipline, leadership and teaming.
- Process Improvement principles: develop and continually improve product life cycle activities so

that they are integrated and occur concurrently whenever possible.

- Information Management principles: higher levels of information and knowledge integration, enhancement of information and knowledge communication, and management of corporate resources (people's knowledge, information technology, etc.).

The application of the above principles will help to ensure that all aspects are considered during the implementation and operation of SE. The extent to which each of the principles is adopted does however tend to vary depending upon the nature of the organisation.

Leading manufacturing companies have began to be aware of the immense benefits that can be derived from SE, however, industry is often forced to adopt a pragmatic approach to simultaneous engineering in order to solve their more immediate problems. The use of advanced information technology solutions seems to play a minor role in the introduction of SE within industry. Organizational issues take priority (Dunn 1990, Siegal 1991, Woodgate 1991, Wheeler et al. 1991), followed by the use of formal methodologies, such as design for assembly and manufacture (Hiatt 1990, Miles 1990, Nichols 1991, Gerhardt et al. 1991, Lee–Mortimer 1991, Booty 1991, Eversheim and Gross 1991), and a range of quality engineering techniques for managing complex system trade–offs and for finding optimum design and production process parameters (Henshall 1989, Denton 1990, Potts 1990). However, this is the current situation and it is likely that as companies become more experienced in SE, they will start to look for more sophisticated information technology tools.

Academic institutions and research laboratories are commonly divided between providing organisational and technological support for major change initiatives. The former research has concentrated on developing methodologies for the introduction of both new organisational structures and team work. Research effort dedicated to defining methodologies for SE introduction can be found in Evans (1990), Gillen and Fitzgerald (1991), Karandikar et al. (1992), and Evans et al. (1994). The nature of this work is related to organisational, cultural and technical issues.

The technology based academic initiatives have focused largely on developing software applications to support the implementation of specific process improvement techniques and in developing frameworks that allow the capture and sharing of cross–functional information. The goals of such information technology research are long–term and are intended to enhance the understanding of future SE support techniques and to smooth the transgression from current information technology solutions to those of the future. A comparison of approaches to concurrent engineering is given in (Dowlatshahi 1994).

The authors believe that to achieve integrated environments for the support of Simultaneous Engineering it is necessary to define and develop Information Models, integrate and implement Decision Support Applications and provide an adequate Information System Architecture. The development of these key technological requirements should be based on frameworks which enable the computer system to be defined, configured and implemented according to the requirements dictated by the enterprise integration strategy. These, in the authors' opinion, are the core elements of future integrated environments for Simultaneous Engineering.

In section 2 a more detailed description of the relevant research that has been undertaken in these areas is presented. Section 3 draws conclusions from the review whilst the appendices contain individual reviews of the systems referenced in the main body of the text. The appendices have been sub–divided so that Appendix A represents stand–alone systems and Appendix B integrated systems.

2. Computer Aided support for Simultaneous Engineering

A broad range of technology based research related to SE has been undertaken by academic institutes. The literature reviewed has highlighted several key technological requirements that must be addressed in order to develop adequate enabling technologies :

1. Modelling Methodologies: a diversity of methodologies are needed to model life cycle information, design and manufacturing processes, enterprise behaviour and organization. In addition, methodologies should tackle the problem of modelling human behaviour in order to better understand the relationships between humans and technology.

2. Computer Aided Decision Support: applications which enable groups and teams to interact, collaborate and make decisions

during product realization are needed. These computer aided decision applications should support the different activities throughout the life cycle of a product e.g. design evaluation, manufacturability analysis, process plan generation, etc. Artificial Intelligence paradigms seem to provide the means to develop effective decision support software tools.

3. Information Architecture: open and distributed computer-based information architectures are required to provide different services for the achievement of high levels of integration and communication, and common interface structures.

4. Frameworks for SE Environment Development: there is a need for frameworks that organize the SE system development process itself. Such development frameworks serve as reference models to establish the time–ordered application of people, methods and tools to define, develop, implement and evolve the SE environment towards the desired level of integration and automation.

This classification is similar to some elements of the six level classification proposed by Kahaner and Lu (1993). In the following sub–sections a more detailed description of these research issues is presented.

2.1. Modelling Methodologies

Methodologies have been developed for modelling the important aspects of the product's life cycle. The research reviewed seems to have concentrated on the use of methodologies for modelling the information required during the product's life cycle i.e. product and manufacturing information.

The kinds of information required for Simultaneous Engineering must be understood before an adequate definition of the nature of computer aided support for Simultaneous Engineering can be created. The availability of relevant, reliable and consistent product and manufacturing information at any time is key to the computer systems which support SE. These systems require the capability to share common sources of information throughout the different phases of a product's life cycle.

The information required throughout the life cycle of a product can be described, structured, stored without redundancy and standardized if the relevant data concerning a company, its products and manufacturing facilities, is defined in Information Models (Weck et al. 1991). These models can be developed using information modelling languages such as EXPRESS (ISO CD 10303 – 11).

The authors believe that Information Models for Simultaneous Engineering ideally should:

- capture and represent product information, and manufacturing process and resource information.
- provide immediate access to information about previous product 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 being developed.
- keep data to be shared by team members in commonly accessible data bases.

The research reviewed related to information modelling for the support of SE has been primarily concerned with:

1. Product Modelling: the modelling of information associated with a product and its components throughout its life–cycle.

2. Manufacturing Modelling: information modelling of manufacturing processes and resources in an enterprise.

There is no one universally agreed definition of the scope and nature of product modelling or product models, however, a selection of definitions are given below:

"Various kinds of technological information are represented in product models .... basic models are used for describing basic properties and constraints about objects ..... solid models for shape representation, symbolic mathematical formula manipulation for describing various constraints, FEM mesh models ...... engineering models ...... can represent the basic engineering knowledge, such as dimensioning and tolerancing, assembling and kinetic relations, material and manufacturing methods, etc.... and application models are models of products currently designed", (Sata et al. 1985).

"Product modeling refers to the activities related to representing and utilizing information related to products, their design and manufacturing processes, and their production management", (Mantyla 1989).

"Product modelling is a modelling framework which can capture and represent all the necessary product information through the whole life-cycle of our products, from initial product planning until maintenance", (Kimura and Suzuki 1989).

"Product models can be interpreted as the computer internal logical structured information in a factory, which is available about a product. This information is integrated into one logical context", (Spur et al. 1989).

"The idea of an integrated product model bases on the approach to describe all necessary information of the different phases of the product life cycle, their interrelation and product dependant views on the product", (Grabowski et al. 1989).

"Product modelling, as an essential part of the Computer Aided Simultaneous Engineering Systems, produces product data models. A product data model, which is generated and instanciated during the product development activity, should be able to support the whole product life cycle concerns" (McKay and Bloor 1991).

"software representation of the form and content of the data that describes a product throughout its life-cycle." (Young and Bell 1992).

"The Product Model covers all the information belonging to the part. These are basically geometric information and the representation of the part", (Bjorke and Myklebust 1992).

"Product information model is a simplified description of facts and concepts about a class of product" (ISO CD 10303 – 1).

"... the term product model can then be interpreted as the logical accumulation of all relevant information concerning a given product during the product life cycle", (Krause et al. 1993).

The Product Modelling concept aims to achieve the integration and sharing of data/information among different engineering activities in an enterprise by the use of a Product Model. Product Model research has been carried out by academic research institutes in order to establish product modelling environments and experiment with the integration of automated applications. Such research has been undertaken by:

1. Tokyo University (Sata et al. 1985, Kimura and Suzuki 1986, Kimura et al. 1987, Kimura and Suzuki 1989).

2. Berlin Technical University (Spur et al. 1986, Spur et al. 1989).

3. University of Karlsruhe (Grabowski et al. 1989).

4. Concurrent Engineering Research Center (Kinstrey et al. 1990).

5. Loughborough University of Technology and Leeds University (Shaw et al. 1989, McKay and Bloor 1991, Young and Bell 1992, Corrigall et al. 1992)

6. IMPPACT, ESPRIT No. 2165 (Bjorke and Myklebust 1992).

7. University of Illinois at Urbana-Champaign (Lu 1992).

8. University of Massachusetts at Amherst and University of Karlsruhe (Nnaji et al. 1993).

Much of the work undertaken by these groups has resulted in computer-based environments that have evolved to support Simultaneous Engineering. In these systems, it has been recognised that it is important to provide a comprehensive frame for dealing with Product Models which can represent, transmit, manipulate and store all the technological information associated with the design and manufacture of the product. All systems reviewed in this paper, have adopted a common source of information to support both applications and users. Some of them have extended this idea to support the representation of knowledge, however, the ways in which this has been achieved are varied (Sohlenius 1992).

Reddy et al. (1993) use a Product, Process and Organization Model (PPO), Wu and Choong (1992) have developed a global-local Product Data Model scheme, Lu (1992) bases his work on Decision Models, IMPPACT (Bjorke and Myklebust 1992) uses an integrated Product and Process Model, Kimura (1993) considers Product, Process and Activities Models, MOSES (ACME/SERC 1991) defines Product and Manufacturing Models, and Cutkosky et al. (1993) describe a design model composed from shared design-domain ontologies, whilst Finger et al. (1992) share a multi-level, dynamic, domain neutral representation of the product design. The kernel of these systems can be considered to be these information models. These models are a repository of information and provide a means by which concurrency in design may be achieved. Other parallel work envisages the need to have a common source of information but it is used as a means of representing specific perspectives of the design process rather than being the kernel of the system. These models support the information requirements of particular applications, however, they are not intended to fully support all life-cycle information needs (e.g. Hambaba et al. 1992, Vujosevic and Kusiak 1991, Cutkosky et al. 1989, Ishii 1992, Bowen and Bahler 1993, O'Grady and Young 1992, Chen et al. 1994).

The work on representing product related information, especially geometry definition, is relatively mature and most research groups have loosely based their product models on the evolving STEP standard (ISO CD 10303 - 1). Nevertheless, various groups are creating their own extensions to the standard in order to improve its utility e.g. Wu et al. (1992) added a Joint Component to the product definition to deal with the assembly problem and MOSES is dealing with the representation of assembly relationships and specifications (Henson et al. 1993).

The rapid development of computer aided information technologies has triggered situations where a set of information models (such as factory models that include tool or material flow models, order models, usage models and feedback information) all have to be linked together (Krause et al. 1993). The emphasis of information modelling is therefore shifting to include models other than those that represent product data.

It is accepted that a Product Model captures the information related to a product throughout its life cycle. In the authors' opinion, to successfully support SE there is a need for another information model to describe and capture the information about the manufacturing processes and resources which are required for product realization e.g. tools, fixtures, machining cells, Flexible Manufacturing Systems, NC-programs, operators, etc. Manufacturing Modelling represents and captures information related to manufacturing processes (characteristics, capabilities, etc.) and manufacturing resources (characteristics, functionality, capabilities, constraints, behaviour, etc.) in order to support design and manufacturing functions.

An early reference to Manufacturing Modelling is to be found in the IMPPACT project documentation (IMPPACT 1991). IMPPACT defines a Factory Model which "structures the information for the facilities such as machine tools, jigs and fixtures, tools, robots, etc." and a Process Model which "consists of information describing the production activities such as processes, operations and pathes".

Kimura's (1991) Manufacturing Resources Model "tries to represent the whole factory ...... it will include the following items: machine tools for various manufacturing processes, tools, fixtures, jigs, control devices, communication equipment, materials, buildings, and human resources, etc. ".

A Facility Model described in Molina et al. (1992) to "represent the production and material handling systems, and the relationships between them according to a specific Flexible Manufacturing System" is used for the design and evaluation of Flexible Manufacturing Cells in a concurrent environment for FMS Design.

The Manufacturing Model has been defined by Al-Ashaab and Young (1992) as the model that "captures the information which describes the characteristics, or behaviour, of the process and the knowledge and constraints which govern the use of the process". This is the foundation for the evolving concept of the MOSES project (Ellis et al. 1993, Molina et al. 1994a).

Project 2 of the ISO/TC184/SC4/WG8 (ISO TC184 SC4 WG8/N13) is related to Resource Usage Management with the scope to "develop generic and according implementation oriented standards that enable enterprises to document resources and entire manufacturing processes, to communicate internally and externally about them and to optimise their Resource Usage Management". This project collaborates with other standardisation bodies to develop an application oriented representation of Resource Usage Management information and function.

The Manufacturing Cell Operator's Expert System (MCOES) uses a Factory Model to represent the information regarding static factory entities and process models (Mantyla 1993b). In this system, a Factory Modeler allows a manufacturing engineer to store static factory resource information in the Factory Model.

The ESPRIT Consortium AMICE (ESPRIT Project 688/5288) has defined the CIMOSA open system architecture to "enable enterprise modelling for efficient exploitation of the enterprise knowledge". CIMOSA has a component that is concerned with information modelling. The information view allows the modelling of information about products, process, resources, organization and business (administrative).

The trend in manufacturing modelling definitely seems to be moving away from the static representation of resources to more useful models which define the capabilities of manufacturing processes and the ways in which resources are used to achieve these processes. Consideration is also now being given to dynamic aspects of manufacturing modelling. The value of using an information model as a basis for scheduling and planning has led to the investigation of how best to include time dependences and event sequences.

Both product and manufacturing models are needed to support information needs during the product life cycle activities. Product modelling is a mature area of research which has led to the development of concepts which are key to effectively providing product information support for simultaneous engineering e.g. Product data exchange standards – STEP (ISO CD 10303 – 1). Manufacturing modelling is a new research area which intends to address the problem of representing and capturing manufacturing information. This will allow the provision of reliable manufacturing information to assist in the performance of life–cycle activities and related decisions. The development of Manufacturing Models has motivated researchers to investigate further the concept of Enterprise Modelling (Eirich 1992).

2.2. Computer Aided Decision Support

Decision support is provided by computer based information systems in order to assist people engaged in decision making activities. Computer Aided Decision Support for Simultaneous Engineering has its roots in applications which were developed to enhance the product design, based on process improvement principles such as design for assembly (Boothroyd 1985), design for manufacture (Stoll 1988, Swift 1987) and design for injection molding (Huh and Kim 1989, Poli et al. 1992, Irani et al. 1989). These sort of applications are discussed in Ettlie and Stoll (1990), and Corbett et al. (1991).

A new generation of enhanced software tools that tackle specific design and manufacturing problems, whilst considering product life cycle concerns, have been created e.g. plastic part design (Gadh et al. 1989, Ishii 1992, Hambaba et al. 1992), process planning (Cutkosky et al. 1989, Lu 1992), printed wiring board design (O'Grady et al. 1991, Bowen et al. 1993) and mechanical design (Rehg et al. 1988, Finger et al. 1992, Wu et al. 1991, Ellis et al. 1993).

These computer systems have been developed as stand–alone applications (Rehg et al. 1988, Gadh et al. 1989, Ishii 1992, Bowen et al. 1993, O'Grady et al. 1991, Hambaba et al. 1992), or as part of integrated environments (Cutkosky et al. 1989, Lu 1992, Wu et al. 1991). The development of the latter environments is usually based on open and distributed computer–based architectures which provide different integration services and allow the communication and exchange of information among the decision support applications.

The stand–alone engineering applications can be considered the first generation of SE applications. These engineering applications are aimed at specific tasks, but they are not part of an integrated environment. These sort of applications improve the product design by concurrently considering different aspects of the life cycle. Examples of such of systems are: object–oriented framework for CE (Vujosevic and Kusiak 1991), life cycle engineering applications (Ishii 1991, 1992, Ishii and Mukherjee 1992), Computer–Aided Simultaneous Engineering system (Rehg et al. 1988), ManuFEATURE (Gadh and Prinz 1992), Design Fusion system (Finger et al. 1992), GALILEO2 (Bowen and Bahler 1992, 1993), SPARK (O'Grady et al. 1991, Young et al. 1991, O'Grady and Young 1992) and the Intelligent Hybrid System for the design of injection moulded parts (Hambaba et al. 1992).

Most of these applications use artificial intelligence techniques to capture life cycle information. Blackboard architectures are common to many of the systems analysed and are used to control the development process and share information among processes which assist the design activity. The artificial intelligence techniques used for information modelling include knowledge representations (Gadh et al. 1989), semantic nets and constraints rules (Regh et al. 1988, Ishii 1992), and constraint nets (Bowen and Bahler 1991, O'Grady and Young 1992). Blackboard architectures are used to control and manage the design process (Vujosevic and Kusiak 1991, Lu 1991, Finger et al. 1992). In general, object oriented implementation methods are used to facilitate the development of such systems.

Some researchers consider the simultaneous engineering problem to be one of satisfying constraints. In order to develop decision support applications they have found it necessary to develop languages for the representation of constraints (O'Grady et al. 1991, Bowen and Bahler 1992, Medland 1993).

Decision support applications can exist in insolation, or within an integration environment, in order to support a small element of the simultaneous engineering activity. They offer significant advantages for the improvement of such specific tasks. These applications are intended to assist the human decision makers to exercise judgement. It appears that some of the systems reviewed have extended this concept so that the system is capable of making decisions with minimum human intervention. In the opinion of the authors this is dangerous unless the tasks assigned to the system are mundane and completely understood. The complex interaction of tasks, and hence lack of structure, that characterises SE limits the number of such tasks.

2.3. Information System Architectures

There has been considerable research and development activity to establish standard Information System Architectures for large-scale engineering and manufacturing businesses. System architectures are regularly employed to help simplify the development task and structure working methods. Only work undertaken into developing Information System Architectures related specifically to simultaneous engineering is relevant to this review (e.g. Sriram et al. 1990, Genesereth et al. 1992, Lu 1992, Reddy et al. 1993, Cutkosky et al. 1993, ESPRIT Projects). Other endeavours to develop such architectures include IISS (Judson 1986), IDS (1989), CFI (Painter 1990), EIS (1990), CIM-BIOSYS (Leech et al. 1991, Weston 1993) and Urban et al. (1994), however, these architectures are not specifically designed for the support of simultaneous engineering and will therefore not be considered further in this review.

Contemporary tools for the support of design include geometric modelling and design analysis applications. Manufacturing applications also currently generate manufacturing information such as process-plans, NC-Code, schedules, etc. The aim of current research is to integrate these well developed applications within an information system architecture, together with new software tools based on SE principles and software environments that support team work.

The Information System Architectures to support SE consist of applications that are integrated within a framework and make use of the integration services of the information system architecture to access information models (e.g. Product Model and Manufacturing Model). Examples of such research architectures are:

- DICE – Distributed and Integrated environment for Computer-aided Engineering (Sriram et al. 1990).
- PACT – Palo Alto Collaborative Testbed (Genesereth et al. 1992, Cutkosky et al. 1993).
- DARPA DICE and CE Testbed (Londono et al. 1990, CERC 1993).
- SWIFT – System Workbench for Integrating and Facilitating Teams (Lu 1992, Lu et al. 1993).
- ESPRIT Projects (CIM-OSA Project No. 688/5288, CMSO Project No. 2277, IMPPACT Project No. 2165)

A study of these architectures has enabled the authors to compile a list of the requirements for an architecture to support SE :

1. Identify, coordinate and communicate between the different perspectives involved in SE, whether represented by groups of people or applications.
2. Provide information and knowledge sources that are able to represent evolving expertise and product designs, that can be readily modified, and that are easily accessible.
3. Monitor the history of the design process so as to enable future design procedures to capture best practice and to maintain accountability.
4. Control and configure the various system elements in a way that is transparent to the user and ensures system integration.
5. Provide an interactive, multimedia interface for the system user.

Information system architectures are essential if teams are to be supported. This type of architecture allows the integration of both information models and decision support applications in a structured and transparent manner. The emphasis in this work is on defining the integration environment and the high level communication protocols between the system elements for the support of simultaneous engineering. The integration environment enables heterogeneous engineering tools to be effectively shared among team members. High level communication protocols allow knowledge sharing for effective teamwork and facilitate the human interaction by integrating multiple communication media such as voice, text and graphics. The ultimate aim is that these architectures can be employed to develop systems that will support designers in a concurrent engineering environment.

The use of information system architectures to structure the development of SE support systems is gaining acceptance. Nevertheless, there is a need for a governing set of guide–lines that will illustrate the potential interactions between different information system architectures. For information systems (the essence of SE support systems) the Reference Model for Open Distributed Processing Systems (RM–ODP) looks to hold promise (Linington 1992, ISO/IEC JTC1/SC21/WG7 N 755).

2.4. Frameworks for SE Environment Development

The terms framework and architecture have been used ambiguously within the manufacturing domain to denote reference models that assist in the development of integrated systems during different phases of, or throughout, the complete life cycle. Mayer and Painter (1991) pointed out the difference between a framework and an architecture. The latter only denotes the information system architecture in terms of databases, networks, operating systems and integration utilities required for the enterprise integration. The framework refers to an organized representation of characterized situation types that occur during an information system life cycle for enterprise integration. Each situation specifies tasks, methods and tools which can be used to support a particular development situation (Zachman 1987).

Based on the characteristics defined by Mayer and Painter (1991), the authors' belief is that a framework for SE environment development should assist the developers in the following three tasks:

1. Identify the enterprise information system requirements to support Simultaneous Engineering i.e. functionality, information, resources, role in the organization, etc.

2. Guide the design and implementation of the information system itself.

3. Organize the establishment of the time–ordered application of people, methods and tools to evolve the SE system towards the desired level of integration and automation.

The use of frameworks for enterprise integration e.g. CIMOSA (*ESPRIT* Project 688/5288), GRAI–GIM (Doumeingts et al. 1992) and Purdue Enterprise Reference Architecture (Williams 1991) has been proven useful in the complex task of describing an integrated system, its life cycle and the methodology for its application (William et al. 1993). This is because those architectures are used as a reference for the definition of situations and methods that can exist, and be used, in the development of integrated enterprise systems. In the authors's opinion these frameworks can be used to assist in the three tasks listed above. Nevertheless, due to their coverage, they combine elements not required for a SE system (e.g. control of manufacturing technology) hence making their use a difficult and complex task.

On the other hand, the research into the development of computer support for Simultaneous Engineering has embraced issues related to distribution and sharing of information, integration of application by means of networking and multimedia environments, and the provision of services for team co–ordination (Kahaner and Lu 1993). These kind of systems are open in nature to allow the incorporation of a wide range of technologies, and distributed, to enable the interaction among remotely located people. An emerging well documented standard related to these issues is the one concerned with open distributed systems, known as the Reference Model for Open Distributed Processing or RM–ODP (Linington 1992, SC21/WG7 N 755). The authors believe that the use of the RM–ODP could be used to guide the design and implementation of Computer Aided Simultaneous Engineering Systems.

An approach being taking by MOSES (Molina et al. 1994b) is to combine RM–ODP and CIMOSA. The RM–ODP is used as the basis to define and create a Reference Model for the development of CAE which supports Simultaneous Engineering. This CAE Reference Model is set into the context of the CIMOSA requirements definition model to enable different enterprises to assess the ways in which a particular CAE system could provide support to their business. This enables the successfull support of the first two tasks i.e. identification of requirements and guidelines for system development.

Mantyla (1993a) presents a hypothetical system architecture called the Open Architecture Concurrent Engineering Framework which embraces the realization of the first two tasks described above based on the following functional subsystems: modelling tool, model–to–configuration compiler, configuration tool and concurrent engineering toolkit.

However, the organization and management of the timely incorporation of people, methods and tools to evolve a SE environment towards the desired level of integration and automation requires further research.

3. Conclusions

This review paper has demonstrated that significant effort is being concentrated on developing computer aided simultaneous engineering systems. The enabling technologies for the support of simultaneous engineering have been classified as either stand–alone applications or integrated systems in Appendices A and B respectively. The general system characteristics are summarised in Table 1 and the implementation characteristics are detailed in Table 2. Stand–alone applications address specific problem domains and aim to improve productivity within these domains e.g. Ishii 1991, Finger et al. 1992, Bowen and Bahler 1993. Integrated systems aim to provide a foundation which will be capable of supporting all elements of product's life cycle e.g. Lu 1992, Cutkosky et al. 1993, Reddy et al. 1993, etc.

The scope for research is considerable, however, in the authors opinion the literature has highlighted several key topics that must be addressed in order to enable the realisation of such systems. These being:

1. Information Modelling
2. Computer Aided Decision Support
3. Information System Architectures
4. Modelling Frameworks

The approach employed for structuring information within simultaneous engineering systems depends upon whether the information is considered as the kernel of an integrated environment (Reddy et al. 1993, ACME/SERC 1991, IMPPACT 1991, Kimura 1991) or purely as support for a specific software application (Finger et al. 1992, Bowen and Bahler 1993, O'Grady and Young 1992). In the former, international standards are employed in order to ensure the extensability of the systems and define the format of the information models e.g. STEP, EXPRESS. The latter approach tends to represent information in a format that is most suitable for a specific application domain e.g. constraint networks.

The provision of a standard format for the representation of information is essential for systems that intend to share information between a diverse range of applications and users (CERC 1993). This need will prompt the adoption and development of standards such as STEP. The evolving STEP standard has influenced how many of the research groups involved in developing integrated frameworks have structured their product data (Wu et al. 1992, Brandt and Petro 1992, ACME/SERC 1991).

The need to have a consistent source of manufacturing data has been identified (IMPPACT 1991, Kimura 1991, Mantyla 1993, Molina et al. 1994a). However, the diversity between the groups involved in manufacturing modelling is great. The scope extends from those that represent only the capabilities of a specific process to those that aim to generate detailed models of the whole manufacturing operation of an enterprise. As yet a framework to position each of these efforts does not exist. A standardisation effort in this area is being undertaken by the ISO TC184 SC4 Working Group 8, known as MANDATE [MANufacturing DATa Exchange].

Future research into information modelling for the support of simultaneous engineering will, in the authors opinion, identify a range of models that represent the informational needs of the extended enterprise e.g. customer, supplier and distributor information. The need for standardised methods of data modelling will become apparent when such a diverse range of models are integrated so as to enable them to be employed by decision support applications.

The current state of decision support applications is such that specific engineering tasks are being addressed effectively ((Bowen and Bahler 1992, Finger et al. 1992,O'Grady et al. 1991, Lu 1992). The area that requires further attention is that of co–ordinating the simultaneous engineering task. This task is difficult because no two organisations seem to share a common understanding of what constitutes simultaneous engineering. Company culture plays a key role in determining how an organisation addresses the implementation of the SE philosophy and hence its' computer aided support.

Information system architectures provide the means to integrate and co-ordinate distributed software applications. This is essential for the implementation of an environment for the support of teamwork (Reddy et al. 1993, Cutkosky et al. 1993). The architectures reviewed are providing a new set of services which facilitate tool integration, distributed access to different applications, multimedia, the sharing of applications and distributed information access. The trend is to provide the means to exchange not only information but knowledge (Lu 1992 and Genesereth et al. 1992)

It is the opinion of the authors that the incorporation of decision support applications and information models into integrated environments will significantly improve the effectiveness of engineering design teams.

At present no flexible approach exists for tailoring the response of systems according to specific enterprise requirements. It is the authors' belief that a framework for simultaneous engineering is needed to alleviate this problem. The framework must include elements of enterprise modelling e.g. function, behaviour and organisation. A framework will allow the requirements of the system to be identified and matched to key system elements. It will also allow the impact of the system on the organisation to be predicted.

All of the systems aim to support simultaneous engineering and it is recognised that they are human centred systems. The importance of team-working and cultural change in the implementation of SE is undiminished, however, it is considered that having systems which are able to provide accurate, timely, cross functional information will encourage such working practices.

ACKNOWLEDGEMENTS

The research is part of an ACME funded project, 'Exploiting Product and Manufacturing Models in Simultaneous Engineering', pursued at Loughborough University of Technology and Leeds University, and supported by a group of industrial collaborators [SERC ref. GR/H 24273 and GR/H 24273 respectively].

The research undertaken by A. Molina is funded by the Mexican Government (CONACyT), Monterrey Institute of Technology (ITESM, Mexico) and ORS Award Scheme (ORS/9226010).

Appendix A
A Review of Stand–alone Simultaneous Engineering Applications

The following text describes stand–alone computer aided support for simultaneous engineering. Stand–alone Simultaneous Engineering applications are the first generation of SE applications. These stand–alone engineering applications are aimed at specific tasks, but they do not support team work. These sort of applications improve the product design by concurrently considering different aspects of the life cycle. The research projects are classified under the laboratories in which they are being pursued.

A1.1 GALILEO3, Bowen J. et al., Computer Science Department, North Carolina State University

Bowen and Bahler (1991, 1992) have demonstrated that object–based constraint networks are a suitable basis on which to build a language for implementing Simultaneous Engineering applications. A constraint defines a relationship that must hold between a group of parameters; a constraint network is a set of constraints which are interconnected by virtue of sharing parameters. In an object–based constraint network, the parameters can be either scalar or complex objects organized in an inheritance hierarchy (Bowen and Bahler 1993). A language named Galileo3 has been implemented, in which a program is a declarative specification of an object–based network and which allows a constraint to be an arbitrary sentence in First Order Free Logic.

Such object–based constraint networks are suitable for representing the artifact being designed and the resources needed to make the artifact (e.g. components, materials, manufacturing environment, etc.) as well as the constraints that could arise due to differences between artifact functionality, component/material properties, and processes throughout the life cycle.

The use of constraint networks in Simultaneous Engineering applications is attractive because of the capability of such networks to propagate information in any direction (multi–directional inference). This characteristic allows an application program to disseminate the restrictiors triggered by decisions made about different phases of the life cycle.

Different engineering applications have been implemented using Galileo3. KLAUS2 (Bowen and Bahler 1992, Bowen et al. 1993) is a system which assists in the concurrent design of printed wiring boards. Other application domains that have been addressed are the design of composite materials (Bahler et al. 1992) and turbine blade design (Bowen 1991).

The research is being extended to explore how Distributed Artificial Intelligence and its paradigms (e.g. multiple agent cooperation) can improve the negotiation between conflicting decisions, while preserving the constraint–based context, on run–time systems (Bowen and Bahler 1993).

A1.2 EDRC, Carnegie–Mellon University

Several research projects related to the computer aided support of simultaneous engineering have been pursued at EDRC.

Rehg et al. (1988) have developed the CASE (Computer–Aided Simultaneous Engineering) system to aid mechanical design and in particular for the problem domain of manual window regulator design. The CASE system is a framework where different program modules interact to solve the design problem. The program modules are classified as: design agents, design critics and design translators. Design agents are programs that manipulate design representations to develop and extend a design according to a group of specifications. The design critics are analysis tools for the evaluation of designs based on certain criteria. Finally, the design translators are programs that map one design representation into another to allow the exchange of design representations between design critics and design agents. The design is represented by a common structure using a semantic network. The structure is composed of a set of parameters imbedded in a constraint network and a set of application rules which represent the conditions for the instantiation of parameters. CASE allows the following tasks to be performed: design synthesis, tolerance generation, interference and clearance analysis and finite element analysis. The software for CASE has been written in Lucid Common Lisp with Portable Common LOOPS, and uses the VEGA solid modeller developed in C and Finite Element software implemented in FORTRAN (Sapossnek et al. 1989).

Gadh et al. (1989, 1991) use a knowledge–based approach that handles features, and feature interactions, in particular focusing on recognition of complex features. The system consists of several experts pertaining to specific elements in the product life cycle e.g. manufacturability. Their main interest is in providing a Manufacturability critic of products created by Net Shape Manufacturing processes such as injection moulding, casting, extrusion and sheet metal-working. The knowledge contained in the manufacturability expert is in the form of rules, based on the part features: their interactions and their parameters, the material used for manufacture and process conditions. One of the experimental systems is PIMES (Plastic Injection Molding Expert System) which uses the extracted features from a CAD system's B-rep solid modeller (Noodles) using a feature graph grammar approach. More recently (Gadh and Prinz 1992) the ManuFEATURE system has been developed to assess the manufacturability of the moulded part. The system recognizes complex shape features using a technique called the Differential Depth Filter.

A computer-based design system is under development (Finger et al. 1992) to enable a designer to consider concurrently the interactions and trade-offs among different, even conflicting, requirements. The system is called 'Design Fusion' and is based on a blackboard architecture that uses a 'heterarchical' control structure. The system architecture provides an interactive environment that facilitates group problem-solving between a designer and knowledge-based systems, and enables the designer to control the available resources i.e. data, knowledge, methods, and algorithms. The architecture has four major components: the blackboard, knowledge sources, search manager, and user interface. The blackboard is used to provide a multi-level, shared, dynamic, domain-neutral representation of the design based on a combination of technologies: geometric representation using B-Rep and CSG, feature representation (graph grammars) and constraints networks. In Design Fusion the focus is on representing the geometry, features, and constraints associated with a design. The knowledge sources are perspectives and methods. The perspectives represent knowledge of different stages in the product-life cycle (e.g fabrication, assembly, testing, etc.). The methods provide standard analysis capabilities such as feature extraction capability, constraint management and mathematical programming. The role of the search manager is to co-ordinate the different perspectives, decide the sequence of contributions and control their execution. The interface allows the user to use the design environment in a completely interactive manner. The system is able to recognize features in an evolving design and to reason with constraints using interval methods and regional partitioning. The first version of the Design Fusion system has been implemented using KnowledgeCraft. The system supports concurrent design applying constraints to enable the simultaneous consideration of different life-cycle elements.

A1.3 Hambaba et al., Intelligent System Laboratory, Stevens Institute of Technology

Hambaba et al. (1992) are developing an intelligent hybrid system that aims to automate the design of injection moulded parts and their moulds by using a combination of conventional AI and neural network approaches. This research integrates these two techniques in a concurrent engineering framework within an object-oriented environment. In this system the plastic part features (slab, boss, rib etc.) are represented as objects and are geometrically linked to each other. The system architecture consists of a tightly-coupled Intelligent Design Model which is itself a rule-based system containing general design rules and neural network models to produce the fuzzy parameters associated with objects. A graphical user interface is provided to enable the user to draw a skeletal design of the plastic part using part features. Missing part features are calculated by the Intelligent Design Model. The design follows the principle of responsibility driven design where the responsibility is given to the most influential feature of the plastic part. This influential feature is designed firstly and then the system designs the remaining features. After the design cycle is completed, the features do a self-check to determine if all their specification has been met. If not the design process is re-iterated until an acceptable design is reached or no further change can be made. This latter situation is considered as a failure due to over specification. This work is to be extended to automate the whole process of designing injection moulded parts including material selection, cost analysis and mould design.

A1.4 Ishii K. et al., Life-cycle Design Lab, Ohio State University

The Life-cycle Engineering Group at Ohio State University (LEGOS) has focused on Design Compatibility Analysis (DCA) in order to achieve concurrent design. Ishii (1991) has addressed in his research the need for a knowledge based program that encompasses knowledge of the various life-cycle elements (e.g. functionality, manufacturability, reliability). Their effort has resulted in the development of a computerised model for design review which utilises DCA. The focus is on identifying the compatibility between the description of a proposed design and the design requirements.

This is achieved by simultaneously evaluating a candidate design from multiple viewpoints, which include not only the functional requirements of a product but also its life–cycle requirements i.e. manufacturing testing, maintenance etc. This kind of system models a typical "round table" meeting where experts evaluate a design from their perspective interests using heuristics or qualitative analysis. The identification of the form (guide–lines/cost formulae), and the development of a systematic method for the application of this knowledge in order to build a computer model for life–cycle design aid, are the main objectives of this research (Ishii and Mukherjee 1992).

A design representation based on semantic nets, using an object oriented representation to capture the life–cycle knowledge, is the core of the research. This representation allows the use of DCA for design evaluation. DCA matches object oriented expressions of compatibility and computes an index in order to suggest improvements. Different design systems have been implemented to demonstrate the benefits of DCA such as: HyperDesign/Service to address the problem of design for serviceability and reliability, HyperQ/Process for process selection in Net Shape Manufacturing and HyperDesign of Plastics, and the HyperGreen information stack for recyclability (Ishii 1992).

A1.5 Kusiak A. et al., Intelligent Systems Laboratory, University of Iowa

A framework for concurrent design is under development (Vujosevic and Kusiak 1991). The system intends to allow the parallel generation of process plans and schedules during feature–based part modelling and to assist the designer in the evaluation of a design from different perspectives. The system consists of the following components: user graphical interface, blackboard data structure (object–oriented data base), problem solvers to assist in the design activities (e.g. feature–based part modelling, process plan generation, manufacturability and schedulability evaluation, etc.), and an assumption–based truth maintenance system to model the dependency relationships between the design and manufacturing information.

The system is being implemented in Smalltalk–80. Some of the knowledge sources used in the problem solving framework have been implemented. Currently, the research effort is devoted to modelling dependency relationships between design, process planning (manufacturing), and scheduling information.

A1.6 SPARK, O'Grady P. et al., Intelligent Design and Manufacturing Laboratory, North Carolina State University

A constraint network language called SPARK has been developed (Young et al. 1991) to implement a wide variety of Simultaneous Engineering applications. SPARK uses frame–based inheritance and is built upon an implementation of first order predicate logic. The predicates allow the representation of a collection of constraints which are interconnected via shared variables that have to be satisfied. Constraint satisfaction is made possible by a combination of user inputs and values automatically determined by the system, thereby avoiding the incompleteness possible with constraint satisfaction problems (O'Grady et al. 1991). The values are propagated bi–directionally through the network. Every time a change is made, in order to preserve the restrictions of the constraint network, a truth maintenance system is used.

SPARK is written in Prolog. The system has been tested on different engineering domain applications such as layout of printed wiring boards, process selection for rotational parts, and turbine blade design (O'Grady and Young 1992).

APPENDIX B
Review of Integrated Frameworks to Support Simultaneous Engineering Team Work

Integrated Frameworks can be considered the second generation of SE applications, in which the support of SE–team work to achieve product realization is enhanced through the use of information models and integrated engineering applications. They are detailed below.

B.1.1 NEXT–CUT, First–Link and PACT, Cutkosky M.R. et al., Centre for Design Research, Stanford University

Cutkosky et. al. (1989, 1991) developed Next–Cut, an experimental computational framework for Simultaneous Engineering at Stanford University. Next–Cut implements the concurrent product and process design methodology which they believe is the best way to achieve design for manufacturability. They suggest that process design should be performed incrementally as the product design is being produced. They propose that if macros of the process operations are associated with design features then process plans can be generated and tested with each new design change. They argue that the combination of features and a process representation is the right foundation upon which to build a complete end–to–end design tool for addressing functional, geometric, and manufacturing constraints. Next–Cut consists of models and modules, which exchange information through a central model which is a knowledge representation, in object oriented format, of the manufacturing environment. Modules are pieces of software consisting of agents (programs have expertise in specific areas) and editors (intelligent graphical tools). The system, which relies heavily on specialized modellers and a planner, aims to create a virtual design team in which users, acting through editors, and computerised agents are interchangeable. The system supports three main manufacturing modes: machining, assembly and injection moulding. The main limitation of Next–Cut is the huge amount of memory which is required to run the system. This a very serious problem which must be resolved before it can be claimed that the system supports concurrent engineering in a manner that can be applied to solve real engineering problems.

Next–Cut supports only single–user operation, however, another two projects (First–Link and PACT) are in progress. They are truly distributed systems and are intended to support team based design. Early results show that they are able to scale up to large problems better than Next–Cut did. First–Link is an agent–based distributed system to support the design and fabrication of wire harness assemblies for aerospace equipment (Park et al. 1992). The architecture, and much of the programming methodology, are derived from Next–Cut. The developers believe that decomposition, abstraction and representation are important issues in providing computational support for simultaneous engineering. First–Link has addressed these issues. The cable design problems are decomposed into five tasks. Each task is supported in First–Link by an agent, containing a set of operations and representations, and an editor for human interaction. Each agent represents an entity that performs a homogeneous set of operations and behaves co–operatively within the structure defined by the framework. The design representation of the system is achieved by using features to support the abstraction. Each feature has to satisfy the cable design domain properties simultaneously. These properties are connectivity, configuration and geometry. The system is currently under development and only preliminary versions of several agents are functioning. These permit designers to construct and modify cable harnesses in response to constraints on connectivity, configuration and geometry whilst automatically taking care of low–level detail.

PACT (Palo Alto Collaborative Testbed) is a concurrent engineering infrastructure that encompasses multiple–sites, subsystems and a range of disciplines (Cutkosky et al. 1993). PACT integrates existing multi–tool systems. The PACT project is aiming to address the following:

1. cooperative development of interfaces, protocols and architecture
2. sharing of knowledge among systems that maintain their own specialized knowledge bases and reasoning mechanisms
3. computer aided support for the negotiation and decision making that characterize concurrent engineering.

PACT is an agent–based architecture system (Genesereth et al. 1992). For convenience, the current PACT version consists of the following four sub–systems, NVisage (a distributed knowledge–based integration environment), DME (Device Modelling Environment, a model formulation and simulation Environment), Next–Cut and Designworld (a digital electronics design simulation, assembly and testing system). The next PACT demonstration will involve other systems. The PACT architecture is an extension to the Open Distributed Processes (ODP) architectures. This extension is achieved by further standardising the style of program interaction by defining message content at three levels:

1. Agent communication is undertaken using the Knowledge Query and Manipulation Language (KQML) of DARPA.

2. Knowledge interchange format is achieved by using KIF language.

3. Using standard vocabularies for representing knowledge about engineering artifacts and processes.

The agents communicate with each other through facilitators which carry undirectional messages to agents that can handle the content of the message. This agent–facilitators arrangement is called a federation architecture. There are six top level agents in PACT. They include digital circuit, control software, power system, physical plant, sensor and parts catalog agents.

PACT has demonstrated good preliminary results on its first experiments which used a robotic manipulator as its subject, a system that combined mechanics, electronics and software. Those experiments were based on low–level message passing between agents. PACT uses a mechanism for distributing reasoning to achieve cooperative work. It provides a framework for knowledge sharing for arbitrary tools. Choosing the right software model for each task is the real issue that PACT has to deal with in order to support complex design at various stages. The next version of PACT will build on commercial substrates such as OMG/CORBA in order to support environments containing thousands of interacting agents. The simulation and analysis services will be transformed into generic engineering applications.

B.1.2 ESPRIT Projects

A range of research projects have been undertaken by this project consortium. Those of relevance to simultaneous engineering have concentrated on developing enabling technologies for Computer Integrated Manufacturing and include the following:

1. CIM–OSA (CIM–Open System Architecture) ESPRIT Project No. 688 is a CIM architecture framework for the design, development and implementation of CIM systems (BSI DD 194:1990)

2. IMPPACT (Integrated Modelling of Product and Processes using Advanced Computer Technologies) ESPRIT Project No. 2165 has developed systems that allow the integration of information pertaining to product design, process and operation planning, including machine control data generation using product and process models (Bjorke and Myklebust 1992).

3. CMSO (CIM for Multi–Supplier–Operations) ESPRIT Project No. 2277 provides an environment for fast and timely communication of both the commercial and technical data that is required for inter–organizational operations (Lischke 1991).

4. CAD*I (CAD Interfaces) ESPRIT Project No. 322 and CADEX (CAD Geometry Data Exchange) ESPRIT Project No. 2195 involve research related to standardised interfaces for CAD systems and the development of the ISO/STEP standard for a neutral file format for CAD data exchange (Bey and Gengenbach 1988).

The results of all the above ESPRIT projects, especially IMPPACT and CMSO, provide enabling technologies that can be used as the basis for successfully implementing Simultaneous Engineering in Europe's enterprises (Lischke et al. 1992).

B.1.3 Kimura F., Precision Machinery Engineering Department, University of Tokyo

A model–based approach is taken by Kimura (1991) for developing manufacturing system software for Product Realization. Product Realization has been defined as "all the necessary activities for converting a given product requirement into a physically realizable object". The focus of this research is on modelling most of the activities of Product Realization, based on product and process models, to create a Virtual Manufacturing environment which simulates real manufacturing. Concurrent Engineering is considered an engineering activity for Product Realization. Computer–aided Rapid Prototyping is used as a complement for Virtual Manufacturing.

In this model–based framework, models are categorized as either computer executable or abstract logical. Object and activity models are computer executable models and are classified according to the criteria of physical constraints. Physically realisable artifacts, manufacturing processes and resources are represented by object models, whilst system and human activities to manipulate these objects are represented by the activity models. There are three object models (Kimura 1993):

1. Product Model: represents every artifact with its related physical realisability constraints and evolving definition (i.e. from ambiguous description to precise specification).

2. Manufacturing Resource Model: this model represents a virtual factory. It consist of models representing real resources that exist in factories (e.g. machine tools for various manufacturing processes, tools, fixtures, jigs, control devices, communication equipment, materials, buildings, and human resources, etc.).

3. Physical Model: a general model that represents physical processes and is used to investigate product functionality and manufacturability.

The activity models are used to represent various kinds of system activities. The object models are classified as:

1. integration model: a meta-model which determines a general system architecture. It serves as a framework to represent other models.

2. design process models: represent all the manufacturing engineering activities such as production planning, product design, process planning, manufacturing preparation, etc.

3. production management model: is a model for managing manufacturing resources in order to optimally allocate manufacturing resources for production.

Kimura (1991) has recognised the importance of the following issues related to the manufacturing software implementation: the system needs to have a modular architecture, the system has to allow adjustments to represent real world status, and comprehensive user-oriented interfaces for human interaction should be considered. This research is not yet fully developed but it seems that better and more efficient software tools based on the idea of Virtual Manufacturing, using object and activities models, can be implemented to achieve a future generation of manufacturing systems.

B.1.4 Lu S.C-Y et al., KBESRL, University of Illinois at Urbana Champaign

Research in the Knowledge Based Engineering System Research Laboratory (KBESRL) at the University of Illinois has focused on the concept of "Knowledge Processing Technology" (KPT) and its impact on Simultaneous Engineering applications (Lu 1991). The main issue of this research is the development of intelligent computer tools for cooperative team support to improve group productivity.

KBESRL has identified that cooperation at the knowledge level is needed in order to fully support Simultaneous Engineering. Cooperation at the knowledge level requires the exchange and sharing of data and knowledge, and two communication modes (one-way batch and multi-way interactive). Therefore, KBESRL research has been mainly concerned with investigating new software technologies and computer tools to enhance and achieve a cooperative team approach at the knowledge level. The challenges that have been identified to address these issues have been categorized as follows (Lu 1992):

1. Integration of complementary engineering expertise: support knowledge sharing and expertise integration during product development (e.g. support multiple data/knowledge representations, integration with CAD and database tools, etc.).

2. Cooperation between multiple competing perspectives: effective management of multiple competing perspectives (e.g. management conflicts, decision histories and rationale, provide comprehensible explanations, etc.).

3. Communication of upstream and downstream concerns: enabling and promoting the communication of decisions at early stages of the product development (e.g. allow early evaluation of decisions, support the least commitment approach in decision making, etc.)

4. Co-ordination of group problem-solving activities: support group productivity by means of group interaction of engineering teams with different expertise and

geographical locations (e.g. use of homogeneous and heterogeneous tools, allow centralised or distributed interactions, etc.)

The computer tools developed using the KPT allow the integration of knowledge, tools and teams. The KPT-based tools can model three types of intelligence (know-how, know-why and know-what) and integrate them to constitute a knowledge-based decision support environment. Tools based on data processing approaches (e.g CAD package, databases, and numerical tools) can be integrated by KPT-based tools due to their capability of allowing data/knowledge exchange and sharing. The integration of teams through the use of KPT-based tools seems possible because data and knowledge provide a more meaningful communication means among team members than pure data sharing.

The following tools have been implemented at KBESRL for different Simultaneous Engineering tasks.

1. D-IDEEA (Distributed Intelligent Design Environment for Engineering Applications) is a major extension of IDEEA. IDEEA is a generic 'know-how' knowledge processing tool and contains various AI tools such as constraints, frames, rules, truth maintenance, composite values and traditional engineering techniques (Herman and Lu 1991). D-IDEEA is aiming to extend the current IDEEA capabilities to support distributed knowledge processing across heterogeneous hardware platforms with synchronous and asynchronous communication between team members (Herman and Lu 1992).

2. AIDEMS (An Interactive Design Evolution Management System) is involved with the implementation of 'know-why' knowledge processing. AIDEMS provides a set of methodologies and tools to manage and utilize decision rationale during design. Design rationale is represented in the form of design plans and design constraints (Thompson and Lu 1990).

3. SWIFT (System Workbench for Integrating and Facilitating Teams) provides a system integration workbench for teams, knowledge and tools. The system facilitates cooperative team activity in a distributed computer environment where complementary expertise and multiple competing perspectives are integrated, and group problem solving activities are co-ordinated (SWIFT 1992).

4. AIMS (Adaptive and Interactive Modeling System) is the core 'know-what' knowledge processing technology. AIMS is a toolbox which integrates machine learning, optimization, and simulation for model based decision support (Tcheng and Lu 1992).

5. LEAD (Learning Assisted Decision Support) is a model-based decision support methodology which provides a toolbox of machine learning algorithms and domain methodologies to assist in domain-specific applications (LEAD 1992).

The above systems form the basis for many other developments. Some engineering applications have been developed using these KPT-based tools, among them are: PRIDE (Tanquary and Lu 1992) an interactive plan organizer and design notebook for individual engineers, CASCADE-T (Wilhelm and Lu 1992) a computer environment for analysis and synthesis of tolerance specifications, and MPEP (Lucenti et al. 1992) a system for manufacturing planing tasks.

KPT-based tools provide the means to achieve an integrated, cooperative and co-ordinated team approach toward the full realization of simultaneous engineering (Lu 1992).

B.1.5 MOSES, Loughborough University of Technology and Leeds University

A computer aided engineering (CAE) system to support simultaneous engineering called MOSES (Model Oriented Simultaneous Engineering System) is being researched by Loughborough University and Leeds University (ACME/SERC 1991). The research is being undertaken in order to define the joint role of product and manufacturing data models in CAE systems of the future. The sharing of common, consistent product and manufacturing data between a range of software applications and design teams is considered key to the effective support of simultaneous engineering. Methods for the identification of conflicts that arise in simultaneous engineering are being explored.

The two data models incorporated in the MOSES research concept are the Product Model and the Manufacturing Model. Each provides a single consistent source of their respective information types. The Product Model contains all data related to a product's life cycle whilst the Manufacturing Model captures all data related to process capabilities, manufacturing resources and the manufacturing strategies employed by the Enterprise (Ellis et al. 1993). The work undertaken by the STEP (STEP 1993) community provides a basis for product modelling, however, additional research into the representation of assembly relationships, product manufacturing information and product specification is being pursued.

The decision support applications that make use of the data models are grouped into a range of application environments. Each application environment contains broad groups of applications that support a specific element of the product life cycle e.g. design for manufacture. The application environments interact with the data models via an integration environment.

An application environment for the support of design for manufacture (DFM) is being developed. The design for manufacture environment at present supports machining and injection moulding. Within each application environment is a Strategist that co-ordinates the operation of a series of experts within its domain based on input from the system user and the content of the product model. The artifacts used to demonstrate the support that the system provides have included the rotors for large electric motors, overhead crane assemblies and tamper-proof plastic food packaging devices. Related research into design for function is being undertaken.

Manufacturing information generation is concerned with supplying information for the manufacture of the product and includes post design activities such as process planning. The role of these post design activities is significantly changed by the incorporation of the DFM environment. Many activities traditionally associated with post design activities are now undertaken by the Design For Manufacture Environment with the resulting product related manufacturing information being stored in the product model.

The system is intended to be extensible so as to enable additional application environments to be included as they are required e.g. Design for Disposal, Design for Aesthetics. A specialist application, called an Engineering Moderator (Harding 1994), identifies conflicts within the Product Model which may arise due to discordant outputs from the different application environments that populate the Product Model. It is this application that helps to co-ordinate the operation of the system and make it suitable for operation in a simultaneous engineering environment.

An experimental system, demonstrating the concepts, has been implemented using an object oriented database and programming techniques. The use of neural networks within the implementation of the Engineering Moderator has been investigated. Implementation prototypes of both a manufacturing model and a product model have been developed using the object oreinted database DEC Object/DB (Molina et al 1994b, McKay et al. 1992). The interfaces to these models have been developed using C++. Earlier research generated a manufacturing model for injection moulding (Al-Ashaab and Young 1992), the prototype of which was implemented using LOOPS, an object oriented environment (Xerox 1988).

The MOSES group are developing a CAE Reference Model in order to define formal methods for the description of system concepts and components (Molina et al. 1994a). This facilitates meaningful discussion and common understanding between system developers and users. The CAE Reference Model is based on the Open Distributed Processing Reference Model (ODP-RM) (Linington 1992).

B.1.6 CERC, Concurrent Engineering Research Center, West Virginia University

The Concurrent Engineering Research Center (CERC) was established as part of the Darpa Initiative in Concurrent Engineering (DICE). Research at CERC has shown that a computer-assisted environment to support concurrent engineering practices requires five generic services (CERC 1992):

1. a shared information model formed by a series of models of product (both form and function), process (both development activities and manufacturing processes), and organization (resources of all kinds). This shared model has been named the PPO (Product, Process and Organization) model (Kinstrey et al. 1990).

2. a networked, multimedia communication environment, using computers to remove barriers of distance, time, and communication between remotely located people and their supporting tools (Srinivas et al. 1992)

3. team co-ordination services that ensures common focus and consistency among people working in parallel (Londono et al. 1990, Nichols 1992).

4. tool and framework integration to provide a standardized collection of facilities for integrating and exploiting application tools (Kannan et al. 1990).

5. management of design history to support continuous improvement based on the rationale of past decisions and best practices.

The main result is the development of a CE Testbed (DICE Architecture) which demonstrates the capabilities and benefits of the DICE technologies. A "parameter-to-part" system centred on a shared model of the product, the process, and the organization has been implemented, together with General Electric Aircraft Engines (GEAE), with focus on the redesign of a hollow airfoil-fan blade (Kamar 1992). Similar scenarios have been implemented in the domain of printed circuit boards, and tubular and sheet metal components for heat exchangers (CERC 1993).

A turbine blade engineering scenario has been implemented using the CE Testbed (Brandt and Petro 1992). In this development the information sharing is provided by different software prototypes such as ROSE, Knowledge Server and PPO server. ROSE (Renesseler Object System for Engineering) is an object oriented database which allows files of objects to be retrieved using the programming languages C++, Object-C and CLOS. The Knowledge Server is a utility that enables users and programs to access information stored in any of its data repositories (e.g. ROSE, PPO server, LASER, SQL server). The PPO server is a shared information repository which maintains a knowledge base of names and versions of existing ROSE application objects. LASER is a commercial frame-based knowledge representation system that can be accessed using a C-language interface. A diversity of engineering software tools have been integrated to assist in the design process: I-DEAS (CAD and FEA package), ICAD (CAD system) and XESS (spreadsheet program).

A software prototype based on the DICE architecture is MONET (Srinivas K. et al. 1992). MONET (Meeting On the NET) is a multimedia virtual meeting system that is designed for Concurrent Engineering applications. Voice, graphics, text, and video, capabilities are available for MONET participants at geographically distributed locations on the network. In MONET, team co-ordination is made possible by using the following systems: Project Co-ordination Board, Constraint Management System, and Requirements Manager. All these systems allow the tracking of a set of resources and the flow of information within a given team. They also provide the design team with the ability to capture, manage, and process constraints during the product design process; and enable the capture and management of product requirements, specifications, and constraints.

"CERC is a comprehensive, resource centre devoted to developing, validating, and disseminating enabling concurrent engineering technologies and evolving standards" (CERC 1992).

B.1.7 MIT DICE, Sriram et al., IESL, MIT

A computer based system called DICE (Distributed and Integrated environment for Computer-aided Engineering) has been developed by the Intelligent Engineering Systems Laboratory at MIT (Sriram et al. 1989). The DICE is targeted to address the coordination and communication problem in engineering, especially in the construction domain. The system can be seen as a network of computers and users, where the communication, coordination and decision support are achieved by the use of a global database, a control mechanism and knowledge modules.

The media for communication is the global database, implemented as a blackboard (Wong et al. 1990). The blackboard contains three types of information: design, negotiation and coordination information. The information related to the product being designed is captured and described in the form of an object hierarchy. The negotiation and coordination information represent the negotiation trace between the various users (i.e. engineers) taking part in the design process and the information required for the coordination of various knowledge modules. The DICE's blackboard can be considered to be an intelligent database. This intelligent database is being implemented using object oriented database technology (Sriram et al. 1990).

Different functions are the responsability of the control mechanism e.g. communication, coordination and data transfer. The major task of the control mechanism is to assist in the negotiation process between the different knowledge modules. In addition, this control mechanism has to evaluate and propagate results of the activities performed by the knowledge modules.

Finally, the knowledge modules are knowledge-based expert systems which assist the control mechanism and the engineers in their decision making. There are four categories of knowledge modules, these being:

strategy, specialist, critic and quantitative. The strategy modules help the control mechanism in the coordination and communication process. For the realization of specialized tasks in the construction process the specialist modules are used. Advise during the product development is given by the critic modules, and the quatitative modules are considered to be a diversity of CAD tools.

An implementation prototype of DICE – called MagpieBridge – has been developed. It is based on GEMSTONE which is a commercial object–oriented database management system. New research work is being undertaken by this research group to explore the following topics : development of knowledge modules to assist preliminary and detailed structural engineering design, improve the constraint management, create a negotiation framework to deal with constraint violation across disciplines, enhance the blackboard, extend the implementation to use other software (e.g. ONTOS and OBSERVER/ENCORE) and provide an multipurpose user interface.

B.1.8 Wu et al., Center for Computer Aided Design (CCAD), University of Iowa

In recent research (Wu et al. 1992), product modelling is used as a basis for realising concurrent engineering of mechanical systems by integrating CAE and CAM applications via a shared product definition using a global–local data model scheme. The Product Model is based on PDES/STEP which provides geometric representation (form features, geometry, precision features) and material properties. A Joint component is added to the product definition to deal with the assembly problem. This PDES/STEP based product definition provides most of the data needed for dynamic analysis, structural analysis, machining process planning, and assembly process planning. Future work will be related with the co–ordination and interpretation of design changes between CAE and CAM applications.

Wu and Choong (1992) have developed a CAE framework based on the DICE–architecture (Londono et al. 1990) to model engineering processes and to capture engineering knowledge for mechanical system design and analysis using an object–oriented approach. The CAE framework implements an integration model which considers three aspects: information integration, control integration and user interface integration. The information integration is related to the common definition and format of product data shared among CAE entities. This information model is based on a global–local data model scheme using PDES/STEP (Wu et al. 1991). The CAE entities are software tools used in engineering activities to develop and analyse products or to manage the procedures of product development and analysis (e.g. Finite Element Analysis tools – ANSYS – NASTRAN – ABAQUS, Geometry Modelling Tools – Unigraphics – I/EMS – PATRAN, Dynamic Simulation Tools – Iowa Driving Simulator). The control integration refers to the protocols governing relations and interplays between CAE entities organized as a tool–workspace–subenvironment–environment entity hierarchy. User interface integration is concerned with providing a unified and consistent graphical user interface. This CAE framework has the basic functionality for cooperative product development. It can be extended in the future to develop a data and process framework for the efficient definition and co–ordination of product data models, process models, and the constraints between these models. This environment is the basis for a new project in Concurrent Engineering for Tracked Vehicles (DARPA BAA91–12).

B.1.9 MCOES, Mantyla , Helsinki University of Technology

Mantyla (1993) describes a hypothetical architecture called the Open Architecture Concurrent Engineering Framework. This framework aims to assist in the definition, configuration and development of simultaneous engineering systems by using the following subsystems:

1. A Modelling Tool which defines the modeling process and its undelying methodology to represent the characteristics of the design and manufacturing environment where the SE system is going to be used and operated.

2. A Model–to–Configuration compiler to create a configuration of the SE system required by the design and manufacturing environment model. The SE system is configurated based on system components and concurrency enabling tools defined in the Concurrent Engineering Toolkit.

3. A Concurrent Engineering Toolkit which describes a diversity of key system components and subsystems which enable the realization of tasks such as: information and knowledge capturing, product

modelling, process modelling, manufacturing system modelling, etc.

4. A Configuration Tool which connects the individual system components by using concurrency enabling technologies such as: communication and synchronization protocols, shared data bases, heterogeneous databases, etc.

These ideas have been applied to create a workshop oriented system called MCOES (Manufacturing Cell Operator's Expert System), for details see Mantyla (1993b).

REFERENCES

ACME/SERC Proposal, 1991, "Exploiting Product and Manufacturing Models in Simultaneous Engineering", Department of Mechanical Engineering–University of Leeds, Department of Manufacturing Engineering–Loughborough University of Technology. UK.

Al–Ashaab A., and Young R.I.M, 1992, "Information Models: An Aid to Concurrency in Injection Moulded Products Design", *Presented in the Winter Annual Meeting '92 of ASME.* Anaheim, California, Nov. 8–13.

Bahler D., Spainhour L., and Bowen J., 1992, "COLADA: A Design Advisor for Composite Materials", Draft Technical Report, Dept. of Computer Science, NCSU, Raleigh, NC, USA.

Bey I., and Gengenbach U., 1988, "The CAD*I Interface for Solid Model Exchange", *Computer and Graphics*, Vol. 12, No. 2, pp. 181–190.

Bjorke O., and Myklebust O., 1992, *IMPPACT: Integrated Modelling of Products and Process using Advanced Computer Technologies*, TAPIR Publishers, Trondheim, Norway.

Boothroyd D., 1985, *Design for Assembly Handbook*, Boothroyd Dewhurts, Inc., Wakefield.

Booty F., 1991, "Approaches to Design for Manufacture", *Works Management*, June 44(6) pp. 57,59,61.

Bowen J., and Bahler D., 1991, "Supporting Cooperation between Multiple Perspectives in a Constraint–Based Approach to Concurrent Engineering", *Journal of Design and Manufacturing*, 1, pp. 89–105.

Bowen J., 1991, "DICE Project Final Report", Dept. of Computer Science, NCSU, Raleigh, NC, USA.

Bowen J., and Bahler D., 1992, "Frames, Quantification, Perspectives, and Negotiation in Constraint Networks for Life–Cycle Engineering", *Artificial Intelligence in Engineering*, 7, pp. 199–226.

Bowen J., Bahler D., and Dholakia D., 1993, "An AI Constraint Network–Based Approach to Bed–Of–Nails DFT for Digital Circuit Design", *Computers Elect. Engineering*, 19(2), pp. 73–85.

Bowen J., and Bahler D., 1993, "Constraint–Based Software for Concurrent Engineering", *IEEE Computer*, 26(1), pp. 66–68.

Brandt W.D., and Petro J.J., 1992, "Integrating the CE Approach of the CERC into Engineering Scenarios", *CE & CALS Conference and Exposition*, June 1–4, Washington, D.C., pp. 203–223.

BSI DD 194:1990, ENV 40 003:1990, "Computer integrated manufacturing (CIM): CIM systems architecture framework for modelling", *British Standard Institution.*

CERC, 1992, "The Concurrent Engineering Research Center", CERC Drawer 2000.

CERC, 1993, "Commercial Off–the–shelf Software for Concurrent Engineering", CERC Report for Task 3.1, Phase 5 of the DARPA Initiative in Concurrent Engineering, FY 1993, *Concurrent Engineering Research in Review*, Volume 5, pp. 15–21.

Chen C., Swift F., Lee S., Ege R. and Shen Q., 1994, "Development of a Feature–based and Object–Oriented Concurrent Engineering System", *Journal of Intelligent Manufacturing, 5, pp23–31.*

Cleetus K.J., 1992, "Definition of Concurrent Engineering", CERC Technical Report Series, CERC–TR–RN–92–003.

Corbett J., Donner M., Meleka J., and Pym C. (Eds.), 1991, *Design for manufacture, strategies, principles and techniques*, Addison–Wesley Publishing Company.

Corrigall M.J., Lee M.K., Young R.I.M., and Bell R., 1992, "Manufacturing Code Generation in a Product Modelling Environment", *Proc. Instn Mech. Engrs.*, Vol. 206, pp. 165–175.

Cutkosky M.R., Brown D.R., and Tenenbaum J.M., 1989, "Extending Concurrent Product and Process Design Toward Earlier Design Stages", *Concurrent Product and Process Design*, N.H. Chao and S.C.Y Lu (Eds.), The Winter Annual Meeting of the ASME, San Francisco, California, pp. 65–72.

Cutkosky M.R., Brown D.R., and Tenenbaum J.M., 1991, "Next–Cut: A Second Generation Framework for Concurrent Engineering", *Computer–Aided Cooperative Product Development*, D. Sriram, R. Logcher and S. Fukuda (Eds.), Springer–Verlag, pp. 1–18.

Cutkosky M.R., Engelmore R.S., Fikes R. E., Genesereth M. R., Gruber T.R., Mark W.S., Tenenbaum J.M. and Weber J.C., 1993, "PACT: An Experiment in Integrating Concurrent Engineering Systems", *IEEE Computer*, January, pp. 28–37.

DARPA BAA91-12, 1991, Solicitation, "Tracked Vehicle Concurrent Engineering Tools Development, Integration, and Validation".

Denton D.K., 1990, "QFD and Taguchi methods enhance productivity and quality", *Industrial Engineering*, May, 22(5), pp. 24–30.

Doumeingts G., Vallespir B., Darricar D., and Chen D., 1992, GIM, GRAI Integrated Methodology, A Methodology for Designing CIM Systems, Version 1.0, Unnumbered Report, LAP/GRAI, University Bordeaux 1, Bordeaux, France, May.

Dowlatshahi S., 1994, "A Comparison of Approaches to Concurrent Engineering", *International Journal of Advanced Manufacturing Technology*, Vol. 9, pp. 106–113.

Dunn J., 1990, "Simultaneous Engineering", *Engineer 11*, October, 271 (7020), pp. 51–53.

Eirich, P., 1992, "Enterprise Modelling: Issues, Problems and Approaches", *Reports of the ICEIMT Special Interest Groups*, Workshop held at the International Conference on Enterprise Integration Modeling Technology (ICEIMT), 6–10 June, Hilton Head, SC.

EIS, 1990, EIS Update Newsletter, 1–3 (Minneapolis MN: Honeywell Inc. EIS Newsletter, published bimontly); EIS *Engineering Information Systems Specifications* (3 vols): vol 1: Organization and Concepts, October 1989; vol. 2: Specifications and Guidelines, October 1989; vol. 3: Engineering Information Model, September 1990.

Ellis, T.I.A., Young, R.I.M. and Bell, R., 1993, "Modelling Manufacturing Process Information To Support Simultaneous Engineering", in N.F.M. Roozenburg (ed.), *Proceedings, 9th International Conference on Engineering Design (ICED'93)*, The Hague, 17–19 August, vol. 3, Edition HEURISTA, pp. 1081–84.

ESPRIT CIM Project 2195, CADEX, CADDETC Presentation, July 1992.

ESPRIT Project 688/5288, AMICE, Volume 1, *CIMOSA: Open System Architecture for CIM*, ESPRIT Consortium AMICE (Eds.), 2nd, revised and extended edition, Springer–Verlag, 1993.

Ettlie J.E., and Stoll H.W., 1990, *Managing the design–manufacturing process*, McGraw–Hill, New York, N.Y.

Evans S., 1990, "Implementation framework for Integrated design teams", *Journal of Engineering Design*, 1(4), pp. 355–363.

Evans S., Lettice F., and Smart P., 1994, "Concurrent Engineering – Key Implementation Issues", Advances in Agile Manufacturing, P.T. Kidd nad W. Karwowski (Eds.), IOS Press, 1994, pp. 89–92.

Eversheim, W. and Gross, M., 1991, "Trends and Experiences in Applying Simultaneous Engineering", in R. Vio and W. Van Puymbroeck (eds), *Proceedings, Seventh CIM–Europe Annual Conference*, Turin, May 29–31, Springer–Verlag, pp. 3–11.

Finger S., Fox M.S., Prinz F.B., and Rinderle J.R., 1992, "Concurrent Design", *Applied Artificial Intelligence*, 6:257–283.

Gadh, R., Hall, M.A., Gursoz, E. L. and Prinz F.B., 1989, "Knowledge Driven Manufacturability Analysis from Feature–Based Representations", *Proceedings, Symposium on Concurrent Product and Process Design*, ASME Winter Annual Meeting, San Francisco, 10–15 December, ASME, pp. 21–34.

Gadh R., Gursoz E.L., Hall M.A., Prinz F.B., Sudhalkar A.M., 1991, "Feature Abstraction in a Knowledge–based Critique of Designs", *Manufacturing Review Journal*, June, 4(2), pp. 115–125.

Gadh R., and Prinz, F. B., 1992, "On the Recognition of Geometric Forms using the Differential Depth

Filter", *Computer Aided Design*, November, 24(11), pp 583–598.

Genesereth M.R., Huyn P.N., and Letsinger R., 1992, "Agent-based Framework for Concurrent Engineering Software", *CE & CALS Conference and Exposition*, June 1–4, Washington, D.C., pp. 135–150.

Gerhardt D.J., Hutchinson W.R., and Mistry D.K., 1991, " Design for manufacture and assembly: case studies and its implementation", *International Journal of Advanced Manufacturing Technology*, May, 6(2), pp. 131–140.

Gillen D.J., and Fitzgerald E.M., 1991, "Expanding knowledge and converging functions", *Concurrent Engineering*, May, 1(3), pp. 20–28.

Grabowski H., Anderl R., Holland–Letz V., Patzold B., and Suhm A., 1989, "An Integrated CAD/CAM–System for Product and Process Modelling", *International GI-IFIP Symposium 89*, Production Technology Centre Berlin, November 8–10, Berlin (West), FRG.

Hambaba M., Elsherif H.E., O'Brien W. and Bergstorm E, 1992, "Mold–base part Design Using Neural Networks", *Intelligent Manufacturing System*: preprints of IFAC workshop. (Eds. R.P. Judd, N.A. Kheir) Oxford, Pergammon, Dearbon, Michigan USA October, pp. 37–41.

Harding, J.A. and Popplewell, K., 1994, "The Rationale for an Engineering Moderator", Moses_report_series_35, MOSES project document, Manufacturing Engineering Department, Loughborough University of Technology, UK, November 1994.

Henshall, E.J., 1989, "Quality engineering", *TQM Magazine*, November, 1 (5), pp 293–298.

Henson B.W., Baxter J.E. and Juster N.P., 1993, "Assembly Representation within a Product Data Framework", Moses_report_series_4, MOSES project document, Mechanical Engineering Department, Leeds University, UK, October 1993.

Herman A., and Lu S.C.Y., 1991, "IDEEA: Intelligent Design Environment for Engineering Automation", *In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory*, University of Illinois at Urbana–Champaign, pp. 49–56.

Herman A., and Lu S.C.Y., 1992, "Computer Methods for Distributed Reasoning", *In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory*, University of Illinois at Urbana–Champaign, pp. 39–46.

Hiatt D., 1990, "Designing for manufacturability: multifunctional teams at Hewlett–Packard", *CIM Review*, 6(4), pp 63–65.

Huh, Y.J. and Kim S.G., 1989, "RIBBER: A Knowledge–based Synthesis System for RIBBED Injection Molded Parts", Concurrent Product and Process Design, (Ed.) N.H. Chao, S.C.Y. Lu., *Proceedings of the Winter Annual Meeting of ASME*, DE Vol. 1, PED Vol. 36, San Francisco, CA, December 10–15, pp. 195–204.

IDS, 1989, "Integrated Design Support System (IDS) AFHRL–TR–89–6: Volume I – Executive Overview; Volume II – IDS Introduction and Summary; Volume III – IDS Requirements; Volume IV – IDS Task Results; Volume V – IDS Software Documentation. AFHRL, WPAFB, OH, December, 1989.

IMPPACT, 1991, "Esprit No. 2165", *Proceedings of the Workshop*, Berlin, 26–27 February 1991.

Irani, R.H.; Kim, B.H. Dixon,J.R., 1989, "Integrating CAE. Features and Iterative Redesign to Automate the Design of Injection Molds", *Proceedings 1989 ASME Int. Computers in Eng. Conference*, Anaheim, California, pp27–33.

Ishii, K. 1991, "Life–cycle Engineering using Design Compatibility Analysis", *Proceedings of the 17th NSF Design and Manufacturing Systems Conference, SME*, January, 1991, Austin, Texas, pp. 1059–1065.

Ishii, K. and S. Mukherjee, 1992, "Computer–aided Life–cycle Design Compatibility Analysis" *Proceedings of the NSF Design and Manufacturing Conference*, SME, January, 1992 , Atlanta, GA.

Ishii, K., 1992, "The Fourth Annual Life–cycle Design Sponsors Meeting", June 9, 1992, Dept. of Mechanical Engineering, The Ohio State University.

ISO CD 10303 – 1, 1992, *Product Data Representation and Exchange – Part 1: Overview and Fundamental Principles.*

ISO CD 10303 – 11, 1993, *Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual.*

ISO TC184 SC4 WG8/N13, Project 2: "Resource Usage Management", Draft Scope presented at the London Meeting by B.R. Katzy, June, 1992.

ISO/IEC JTC1/SC21/WG7 N 755, 1993, "Draft Recommendation X.901: Basic Reference Model of Open Distributed Processing – Part 1: Overview and guide to use", ISO/IEC JTC1/SC21/WG7.

Judson D.L., 1986, "Integrated Information Support Systems", Integrated Information Support System (IISS): An Evolutionary Approach to Integration, Manufacturing Technology Division, Materials Laboratory, Air Force Wright Aeronautical Laboratories, 1985.

Kahaner D., and Lu S., 1993, "First CIRP International Workshop on Concurrent Engineering for Product Realization", *Concurrent Engineering Research in Review*, Volume 5, pp. 6–14.

Kamar J. S., 1992, "Concurrent Engineering Pilot Project at GE Aircraft Engines", *Concurrent Engineering Research in Review*, Autumn, Volume 4, Special Issue, pp. 14–23.

Kannan R., Cleetus K.J., and Reddy Y.V., 1990, "The Local Concurrency Manager in Distributed Computing", *Proceedings 2nd National Symposium on Concurrent Engineering*, February 7–9, Morgantown, West Virginia, pp. 219–239.

Karandikar H.M., Wood R.T., and Byrd J., 1992, "Process and Technology Readiness Assessment for Implementing Concurrent Engineering", CERC Technical Report Series, CERC–TR–TM–92–004.

Kimura F., and Suzuki H., 1986, "Variational product design by constraint propagation and satisfaction in product modelling", *Annals of the CIRP*, Vol. 35, No.1, pp. 75–78.

Kimura F., Suzuki H., Ando H., T. Sato, and Kinosada A., 1987 "Variational geometry based on logical constraints an its applications to Product Modelling", *Annals of the CIRP*, Vol. 36, No. 1, pp. 65–68.

Kimura F., and Suzuki H., 1989, "A CAD System for Efficient Product Design Based on Design Intent", *Annals of the CIRP*, Vol. 38, No. 1, pp. 149–152.

Kimura F., 1991, "Software for Product Realization", *International Symposium for International Trends in Manufacturing Towards the 21st Century*, Oct. 18, Berlin, Germany.

Kimura F., 1993, "Product and Process Modelling as a Kernel for Virtual Manufacturing Environment", Annals of the CIRP Vol. 42/1/93, pp. 147–150.

Kinstrey M., Lee V.K., Lewis J.W., Salant E.L., Sarachan B.D., and Wilson P.R., 1990, "PPO Management", *Proceedings 2nd National Symposium on Concurrent Engineering*, February 7–9, Morgantown, West Virginia, pp. 67–90.

Krause F–L., Kimura F., Kjellberg T., and Lu S.C.–Y., 1993, "Product Modelling", Keynote papers, Annals of the CIRP, Vol. 42/2/1993, pp. 695–706.

Lee–Mortimer A., 1991, "Simultaneous engineering at NCR", *TQM Magazine*, April, 3(2), pp. 99–104.

LEAD, 1992, "Learning Assisted Decision Support (LEAD)", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 113–120.

Leech M.J., Weston R.H., Hodgson A., Clements P., and Ryan A., 1991, "Incremental building of CIM Systems", *International Conference on CIM*, Singapore, September.

Linington, P.F., 1992, "Introduction to the Open Distributed Processing Basic Reference Model", in J. de Meer, V. Heymer and R. Roth (eds), Open Distributed Processing, *Proceedings of the IFIP TC6/WG6.4 International Workshop on Open Distributed Processing*, Berlin, October 8–11, 1991, Elsevier Science Publishers B.V. (North–Holland), 1992 IFIP, pp. 3–13.

Linton L.R., Hall, D., Hutchison, K., Hoffman, D., Evanczuk, S. and Sullivan P., 1992, "First Principles of Concurrent Engineering: A Competitive Strategy for Electronic Product Development", CALS/CE Electronic Task Group, May 1992, CALS Technical Report 005.

Lischke C., 1991, Technical Information Management System, ESPRIT II Project CMSO, Derivable 13, Bremen.

Lischke, C., Kuhlmann, T., Lehmann, F. and Cremer R., 1992, "Inter–Organizational Concurrent Engineering", in C. O'Brien, P. MacConaill and W. Van Puymbroeck (eds), *Proceedings, Eight CIM–Europe Annual Conference*, May 27–29, Birmingham, UK, Springer–Verlag, pp. 167–181.

Londono F., Cleetus K.J., and Reddy Y.V., 1990, "A Blackboard Scheme for Cooperative

Problem–Solving by Human Experts", *Computer–Aided Cooperative Product Development*, D. Sriram, R. Logcher and S. Fukuda (Eds.), Springer–Verlag, pp. 26–50.

Lu S.C–Y., 1991, "Integrated and Cooperative Knowledge Processing Technology for Concurrent Engineering", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign Champaign, pp. 2–23.

Lu S.C–Y., 1992, "Research, Development and Implementation of Knowledge processing Tools to Support Concurrent Engineering Tasks". In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 3–25.

Lu, S. C–Y., Smith, K., Herman, A., Mattox, D., Silliman, M., Lucenti, M., Jacobs, J., Chazin, D., Lawley, M. and Case, M., 1993, "SWIFT: System Workbench for Integrating and Facilitating Teams", *Proceedings, 2nd Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises (WET ICE)*, Morgantown, West Virgina, April 20–22, IEEE Computer Society Press, pp. 48–59.

Lucenti M., Lu S.C.Y., and Ferriera P.M., 1992, "Manufacturing Planning Enabling Platform", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 63–72.

Mantyla M., 1989, "Directions for Research in Product Modelling", *Proceeding of the Third Inter. IFIP Conference on Computer Applications in Production and Engineering*, Tokyo – Japan 2–5 October pp 71–85.

Mantyla M., 1993a, "Towards Open Architecture Concurrent Engineering Frameworks", Otaniemi 1993/TKO–B96, Department of Computer Science, Faculty of Information Technology, Helsinki University of Technology.

Mantyla M., 1993b, "Representation of Process Planning Knowledge for Part Families", Annals of the CIRP, Vol. 42/1/1993.

Mayer R.J., and Painter M.K., 1991, "Roadmap for Enterprise Integration", *Conference Proceedings AUTOFACT'91*, November 10–14, 1991, Chicago, Illinois, SME, pp. 7:1–7:26.

McKay A., and Bloor M.S., 1991, "The role of product models in effective CADCAM", *Proceedings of the Institution of Mechanical Engineers*, Effective CADCAM'91, European Conference, 6–7 November, Coventry, UK, pp. 113–119.

Mckay A., Dawson P.G., Juster N.P., Baxter J.E. and Henson B.W., 1992, "Harmonisation Experiment", Moses_report_series_2, MOSES project document, Mechanical Engineering Department, Leeds University, UK, October 1992.

Medland, A.J., 1993, "The Dynamic Reconfiguration of Rules within a Constraint Modelling Process to Address Evolving Design Problems", in N.F.M. Roozenburg (ed.), *Proceedings, 9th International Conference on Engineering Design (ICED'93)*, The Hague, August 17–19, vol. 3, Edition HEURISTA, pp. 1327–34.

Miles B., 1990, "Design for manufacture techniques help the team make early decisions", *Journal of Engineering Design*, 1 (4) pp. 365–371.

Molina A., Mezgar I., and Kovacs G., 1992, "Object Knowledge Representation Models for Concurrent Design of FMS", *Human Aspects in Computer Integrated Manufacturing*, G.J. Olling, F. Kimura (editors), Elsevier Science Publishers B.V., (North–Holland), IFIP, pp. 779–788.

Molina, A., Ellis, T.I.A, Young. R.I.M and Bell. R., 1994a, "Modelling Manufacturing Resources, Processes and Strategies to Support Concurrent Engineering", in A.J. Paul and M. Sobolewski (eds), *Concurrent Engineering: Research and Applications – A Global Perspective*, Conference Proceedings, 29–31 August, Pittsburgh, Penn, Concurrent Technologies Corporation, pp. 51–60.

Molina A., Ellis T.IA., Young R.I.M., and Bell R., 1994b, "Methods and Tools for Modelling Manufacturing Information to Support Simultaneous Engineering", Preprints, 2nd IFAC/IFIP/IFORS Workshop Intelligent Manufacturing Systems – IMS'94, (Ed. P. Kopacek), Vienna – Austria – June 13–15, 1994, pp. 87–93.

Nevins J.L. and Whitney D.E., 1989, *Concurrent Design of Product and Processes: A Strategy for the Next Generation in Manufacturing*, McGraw–Hill Publishing Company, N.Y.

Nichols, K., 1991, "Simultaneous engineering at DEC", *TQM Magazine*, April, 3(2), pp. 105–110.

Nichols, D.M., 1992, "Monitoring Design Progress in a Concurrent Engineering Environment", *CE & CALS Conference and Exposition*, June 1–4, Washington, D.C., pp. 65–76.

Nnaji B.O., Liu H–C, and Rembold U., 1993, "A product modeller for discrete components", Int. J. Prod. Res., Vol. 31, No. 9, 2017–2044.

O'Grady P., Young R.E., Greef A., and Smith L., 1991, "An advice system for concurrent engineering", *International Jounal of Computer Integrated Manufacturing*, Vol. 4, No. 2, pp. 63–70.

O'Grady P., and Young R.E., 1992, "Constraint Nets for Life Cycle Engineering: Concurrent Engineering", *Proceedings of the National Sciences Foundation Grantees Conference*, January.

Painter, P (Ed.), 1990, CAD Framework Initiative, Architecture Technical Subcommittee: Requirements Task Group, "CAD Framework Users, Goals, and Objectives", Version 0.8, Draft Proposal Document, May 14.

Painter M.K., Mayer R.J., and Cullinane T.P., 1991, "The Many Faces of Concurrent Engineering", *AUTOFACT'91, Conference Proceedings*, November 10–14, 1991, Chicago, Illinois, SME, pp. 17:1–17:25.

Park H., Lee S.H., and Cutkosky M.R., 1992, "Computational Support for Concurrent Engineering of Cable Harness", *Proceedings of the ASME Computers in Engineering Conference*, Vol. 1, San Francisco, August, pp. 261–268.

Poli C., Kuo S., and Graves R., 1992, "Design for Injection Molding: A Group Technology–based Approach", *Journal of Engineering Design*, Vol.3, No. 4, PP. 307–324.

Potts, J.S., 1990, "Choosing the right quality programme", *Quality Progress*, October, 23 (10), pp. 62–64.

Reddy R.Y.V., Srinivas K., Jagannathan V., and Karinthi R., 1993, "Computer Support for Concurrent Engineering", *IEEE Computer*, 26, 1, pp. 12–16.

Rehg J., Elfes A., Talukdar S., Woodbury R., Eisenberger M., and Edahl R., 1988, "CASE: Computer–aided simultaneous engineering", *Artificial Intelligence in Engineering: Design*, (Ed.) J.S. Gero, Elsevier, pp. 339–360.

Sapossnek S., Talukdar S., Elfes A., Sedas S., Elsenberger M., and Hou L., 1989, "Design Critics in the Computer–Aided Simultaneous Engineering (CASE) Project", Concurrent Product and Process Design, (Ed.) N.H. Chao, S C.Y. Lu., *Proceedings of the Winter Annual Meeting of ASME*, DE Vol.1, PED Vol. 36, San Francisco, CA, December 10–15, pp. 137–141.

Sata T., Kimura F., Suzuki H., and Fujita T., 1985, "Designing Machine Assembly Structure Using Geometric Constraints in Product Modelling", *Annals of the CIRP*, Vol. 34, No. 1, pp. 169–172.

Shaw N.K., Bloor M.S., and de Pennington A., 1989, "Product Data Models", *Research in Engineering Design*, June/July, pp. 43–50.

Siegal B., 1991, "Organising for a successful CE process", *Industrial Engineering*, December, 23(12) pp. 15–19.

So T., Jagannathan V., and Raman R.S., 1992, "The Role of Configuration Management Systems in a Concurrent Engineering Environment", CERC technical Report Series, CERC–TR–TM–92–015.

Sohlenius G., 1992, "Concurrent Engineering", Keynote Paper, Annals of the CIRP Vol. 41/2/1992.

Spur G., Krause F.L., and Armbrust P., 1986, "Product Models as Basis for Integrated Design and Manufacturing", *International Journal of Machine Tool Design Research*, Vol. 26, No. 2, pp. 171–178.

Spur G., Krause F.L., and Lehmann C.M., 1989, "Integration of methods for knowledge processing and geometric modelling", *Proceedings of the Institution of Mechanical Engineers*, Fourth European Conference, Effective CADCAM, 6–7 December, London, pp. 57–63.

Srinivas K., Reddy R., Babadi A., Kamana S., Kumar V., and Dai Z., 1992, "MONET: A Multi–media System for Conferencing and Application Sharing in Distributed Systems", CERC Technical Report Series, CERC–TR–RN–91–009.

Sriram D., Logcher R., Wong A., and Ahemd S., 1990, "An Object–Oriented Framework for Collaborative Engineering Design", Computer–Aided Cooperative Product Development, D. Sriram, R. Logcher and S. Fukuda (Eds.), Springer–Verlag, pp. 51–92.

Sriram D., Stephanopoulos G., Logcher R., Gossard D., Groleau N., Serrano D., and Navinchandra D., "Knowledge–Based System Applications in

Engineering Desing; Research at MIT", AI Magazine, Research in Progress, Fall 1989, pp. 79–96.

Stoll H.W., 1988, "Design for Manufacture", *Manufacturing Engineering*, January, pp. 67–73.

Swift, K., 1987, *Knowledge Based Design for Manufacture*, Kogan Page, London.

SWIFT, 1992, "System Workbench for Integrating and Facilitating Teams (SWIFT)", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 29–38.

Tanquary J.W., and Lu S.C.Y., 1992, "PRIME: A Project Organizer and Decision Notebook for Design Teams", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 81–90.

Tcheng D.K., and Lu S.C.Y. Lu, 1992, "AIMS: Integrating Induction and Optimization Algorithms for Model Based Decision Support", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 121–128.

Thompson J.B., and Lu S.C.Y., 1990, "Design evolution management: A methodology for representing and utilizing design rationale", *Proceedings of the 2nd International ASME Conference on Design Theory and Methodology*, Chicago, IL., September, pp. 185–191.

Urban S.D., Shah J.J., Rogers M., Jeon D.K., Ravi P. and Bliznakov P., 1994, "A Heterogeneous, Active Database Architecture for Engineering Data Management", *International Journal of Computer Integrated Manufacturing, Vol. 7 No. 5, pp276–293.*

Vujosevic R., and Kusiak A., 1991, "Data Base Requirements for Concurrent Design Systems", In Engineering Databases: An Engineering Resource, *Proceedings of the ASME International Computer in Engineering Conference & Exposition*, (ed.) V. Saxena, August 18–22, Santa Clara, California, pp. 109–113.

Weck M., Eversheim W., Konig W., and Pfeifer T., 1991, *Production Engineering – The Competitive Edge*, Butterworth–Heinemann.

Weston R.H., 1993, "Steps towards enterprise–wide integration: a definition of need and first–generation open solutions", Int. J. Prod. Res., Vol. 31, No. 9, pp. 2235–2254.

Williams T.J., 1991, The Purdue Enterprise Reference Architecture, Report Number 154, Purdue Laboratory for Applied Industrial Control, December.

Williams T.J., Bernus P., Brosvic J., Chen D., Doumeingts G., Nemes L., Nevins J.L., Vallespir B., Vliestra J., and Zoetekouw D., 1993, "Architectures for Integrating Manufacturing Activities and Enterprises", Information Infrastructure Systems for Manufacturing, Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the Design of Information Infrastructure Systems for Manufacturing, DIISM'93, Tokyo, Japan, 8–10 November, 1993, H. Yoshikawa and J. Goossenaerts (Eds.), North–Holland, pp. 3–16.

Winner R.I., Pennell J.P., Bertrand H.E., and Slusarczuk M.M.G., 1988, "The Role of Concurrent Engineering in Weapon System Acquisition", IDA–Report R–338.

Wheeler R., Burnett R.W., et al., 1991, "Concurrent engineering: success stories in instrumentation, communications", *IEEE Spectrum*, July 1991, 28(7), pp. 32–37.

Wilhelm R., and Lu S.C.Y., 1992, "Tolerance Synthesis to Support Concurrent Engineering", In the Annual Report of Knowledge–Based Engineering Systems Research Laboratory, University of Illinois at Urbana–Champaign, pp. 47–54.

Wong A., Sriram D., and Logcher R., 1990, "User Interfaces for Cooperative Product Development", Proceedings 2nd National Simposium on Concurrent Engineering, February 7–9, Morgantown, West Virginia, pp. 257–283.

Woodgate S., 1991, "Concurrent Engineering: it works, and here is the proof", *Engineering Computers*, 10 (1), pp. 34, 35, 37.

Wu, J.K., F.N. Choong, K.K. Choi, and E.J. Haug, 1991, " Data Model for Simulation–Based Design of Mechanical Systems", *International Journal of Systems Automation: Research and Applications (SARA)*, 1, pp. 67–87.

Wu, J.K., T.H. Liu, and G.W. Fisher, 1992, "An Integrated PDES/STEP Based Information Model for CAE and CAM Applications", Technical Report R–129, Center for Simulation and Design

Optimization of Mechanical Systems, The University of Iowa.

Wu, J.K. and F.N. Choong, 1992, "A Process Modeling System for Concurrent Engineering of Mechanical Systems", *Intelligent Systems in Concurrent Engineering: Proceedings of the AID'92 Workshop*, TR-92-15, North Carolina State University.

Xerox Corporation 1988, *Xerox LOOPS Reference Manual*, Lyric/Medley Release, Xerox PARC, USA, July.

Young, R.I.M. and Bell R., 1992, Machine Operation Planning in Product Modelling Environment, *International Journal of Production Research*, Vol. 30, No. 11, pp. 2487–2513.

Young R.E., Greef A., and O'Grady P., (1991), "Spark: An Artificial Intelligence Constraint Network System for Concurrent Engineering", *Proceedings First International Conference on Artificial Intelligence in Design*, Butterworth-Heinemann Publishers.

Zachman J.A., 1987, "A framework for information systems architecture", IBM Systems Journal, vol. 26, no. 3, 1987, pp. 276–292.

Group	System	Domain	System Characteristics
Bowen et al. CSD	GALILEO3 Based systems	Printed Wiring Boards Design Design of Composite Materials Turbine Blade Design	Single user, description of constraints using first order free logic
Finger et al. EDRC	Design Fusion	Mechanical Design	Single user, graphical interface, feature extraction using graph grammar, reasoning on constraints using interval methods and regional partitioning
Gadh et al. EDRC	PIMES (Plastic Injection Molding Expert)	Injection Moulding	Single user, graphical interface, integration of applications (Solid Modeller–Noodles), feature extraction using graph grammar and differential deep filter
Hambaba (SIT)	Mold–base System	Design for Injection Moulding	Design of plastic parts, graphical interface, feature–based design
Ishii et al. LEGOS	HyperDesign/Service HyperQ/Process HyperDesign/Plastics	Design for Serviceability Design for Reliability Process and Material Selection for Net Shape Manufacture Design for Injection Moulding	Single user, graphical interface use of Design Compatibility Analysis (DCA)
Kusiak et al. ISL		Process Planning Scheduling	Single user, graphical interface, feature–based design
Rehg et al. EDRC	CASE (Computer–Aided Simultaneous Engineering)	Mechanical Design (manual window regulator design.)	Single user, graphical interface, integration of applications (Solid Modeller–VEGA, FEA software)
O'Grady et al. LISDM	SPARK Based systems	Process Selection for Rotational Parts Printed Wiring Board Design Turbine Blade Design	Single User, description of constraints using first order predicate logic

Table 1	SIMULTANEOUS ENGINEERING SYSTEMS: DOMAIN AND GENERAL CHARACTERISTICS	LUT–SE Research Group

Group	System	Domain	System Characteristics
Cutkosky et al. Stanford	NEXT–CUT	Process Planning Assembly Injection Moulding	Single user, graphical interface, feature–based design integration of applications (Solid Modeller–Alpha_1 – Vantage, Process Planner–Propel).
	FIRST–LINK	Cable Harness Design	Team work design, architecture and programming methodology derived from NEXT–CUT
	PACT	Multiple Domain	Team work design, graphical interface, integration of applications (Distributed Knowledge -based Integration Environment–NVisage, Device Modelling Environment–DME, NEXT–CUT and Digital Electronics Design Simulation – Designworld)
ESPRIT	CIM–OSA	CIM architecture framework	Reference Model for design and implementing CIM systems
	IMPPACT	Information Integration among Product Design, Process Planning and NC Code Generation	Team work design, graphical interface, information model oriented system using product and process models
	CMSO	Communication Data Environment	Multiple user, communication network
	CAD*I and CADEX	CAD Data exchange	Standardisation work for STEP
Kimura et al. University of Tokyo		Virtual Manufacturing	Team work design, information model oriented system using object and activities models

Table 1	SIMULTANEOUS ENGINEERING SYSTEMS: DOMAIN AND GENERAL CHARACTERISTICS	LUT–SE Research Group

Group	System	Domain	System Characteristics
Lu et al. KBESRL	D–IDEEA (IDEEA)	Multiple Domain	Distributed knowledge processing tool, integration of AI tools (constraints, frames, rules, truth maintenance and composite values)
	AIDEMS	Multiple Domain	Design evolution management system using design plans and design constraints
	SWIFT	Multiple Domain	Team work design, system integration workbench for teams, knowledge processing and computer tools in distributed computer environments.
	AIMS	Multiple Domain	Integration of learning and optimization algorithms for model based decision support
	LEAD	Multiple Domain	Integration of a variety of machine learning based methodologies with traditional traditional techniques (e.g. simulation, statistics, optimization) for model–based decision support, uses models generated by AIMS.
LUT/ Leeds	MOSES	Machining, Injection Moulding	Team work design, information model oriented system using product and manufacturing models

Table 1	SIMULTANEOUS ENGINEERING SYSTEMS: DOMAIN AND GENERAL CHARACTERISTICS	LUT–SE Research Group

Group	System	Domain	System Characteristics
Reddy et al. CERC	CE Testbed (DICE Architecture)	Multiple Domain	Team work design, information model oriented system architecture using product, process and organisation models, integration of applications (CAD–FEA packages – I–DEAS – ICAD, spreadsheet – XESS, Databases, text and graphical editors), X–windows based GUI interface,
	MONET	Multiple Domain	Team work design, Multi–medi interface (voice, text, video, graphics), Project Coordination, Constraint Management, Resource management, uses the services of DICE architecture.
Sriram et al. MIT	MIT DICE	Construction	Team work design, information model oriented system architecture using design, negotiation and coordination models, integration of four knowledge modules: strategy, specialist, critic and quantitative, X–windows based GUI interface (DICE –UI)
Wu et al. CCAD		Mechanical and Structural Design	Team work design, graphical interface, information model oriented system using product and process models, integration of applications (Finite Element Analysis tools – ANSYS – NASTRAN – ABAQUS, Geometry Modelling Tools – Unigraphics – I/EMS – PATRAN, Dynamic Simulation Tools – Iowa Driving Simulator)

Table 1	SIMULTANEOUS ENGINEERING SYSTEMS: DOMAIN AND GENERAL CHARACTERISTICS	LUT–SE Research Group

Group	System	Information Representation	Information System Architecture/AI Technique	Information Standards	Implementation Technology
Bowen et al. NCSU	GALILEO3–Based Systems	Object Based Constraint Networks	Constraint Networks	N/A	Prolog
Finger et al. EDRC	Design Fusion	Features Graph Grammar Constraints Networks	Blackboard Architecture	N/A	KnowledgeCraft
Gadh et al. EDRC	PIMES	Frame–based Object Oriented	Knowledge Based Rules	N/A	Expert system Shell– Neuron Data's Nexpert Object
Hambaba SIT	Mold–base System	Object Oriented	Neural Networks, and Rule Based System	N/A	C++
Ishii et al. LEGOS	Hyper/X	Object Oriented Semantic Networks	Knowledge Based Rules Constraints	N/A	HyperCard, Prolog
Kusiak et al ISL		Object Oriented	Blackboard Architecture	N/A	Smalltalk–80, OODB
Rehg et al. EDRC	CASE	Semantic Networks Constraints Networks Object Oriented	Cooperative Hierarchical Multi–agents	N/A	Lucid Common Lisp with portable Common LOOPS
O'Grady et al. NCSU	SPARK–Based Systems	Frame Based Constraint Networks	Constraint Networks	N/A	Prolog

Table 2	SIMULTANEOUS ENGINEERING SYSTEMS IMPLEMENTATION CHARACTERISTICS	LUT–SE Research Group

Group	System	Information Representation	Information System Architecture/AI Technique	Information Standards	Implementation Technology
Cutkosky et al. Stanford	NEXT–CUT	Object–Oriented Knowledge Based	Agent Based	N/A	Lucid Common Lisp OO Programming Symbolic Environment
	FIRST–LINK	Object–Oriented Knowledge Based	Agent Based Framework	IGES	Lucid Common Lisp OO Programming Symbolic Environment
	PACT	Object–Oriented Knowledge Based	Agent Based Framework	IGES, KIF, KQLM extension of ODP	N/A
ESPRIT	IMPPACT	STEP/EXPRESS	IMPPACT CIM Architecture	IGES, STEP	C, IMPPACT Shared Database, IMPPACT Data Manager, IMPPACT CIM Modules
	CMSO	EDI	Technical Information Management System	ODETTA, EDIFACT	N/A
LUT/ Leeds	MOSES	STEP/EXPRESS Object–Oriented	RM–ODP	STEP	C++, OODB (Objectivity) LOOPS, Neural Nets
Lu et al. KBESRL	AIMS	Object–Oriented	Induction algorithms Decomposition algorithms Optimization algorithms	N/A	Common Lisp being translated to C

Table 2	SIMULTANEOUS ENGINEERING SYSTEMS IMPLEMENTATION CHARACTERISTICS	LUT–SE Research Group

Group	System	Information Representation	Information System Architecture/AI Technique	Information Standards	Implementation Technology
Lu et al. KBESRL	LEAD	Object–Oriented	machine learning algorithms	N/A	Common Lisp
	IDEEA (D–IDEEA)	Frame–based Composite Data Objects	Constraints–based Language Rules–based Inference Truth Maintenance System Blackboard Architecture	N/A	Lucid Common Lisp Database Interface with ORACLE
	AIDEMS	Frame–based	Blackboard Architecture	N/A	Lucid Common Lisp
	SWIFT	Frame–based Composite Data Objects	Constraints–based Language Rules–based Inference Truth Maintenance System Blackboard Architecture	N/A	C++, Relational Data Base, Mathematica, OODM (ObjectStore)
Reddy et al. CERC	CE Testbed	STEP/EXPRESS Object–Oriented	DICE Architecture	IGES, PDES/STEP	C++, Objective–C LISP/CLOS, ROSE, Knowledge Server, PPO Server, LASER, SQL Server (Oracle)
	MONET	Object Oriented	DICE Architecture	IGES, PDES/STEP	DICE Technologies
Sriram et al. MIT	MIT–DICE	Object Oriented	Blackboard Architecture	N/A	GEMSTONE OODMS, C++
Wu et al. CCAD		Object–Oriented	DICE Architecture	PDES/STEP	DICE technologies

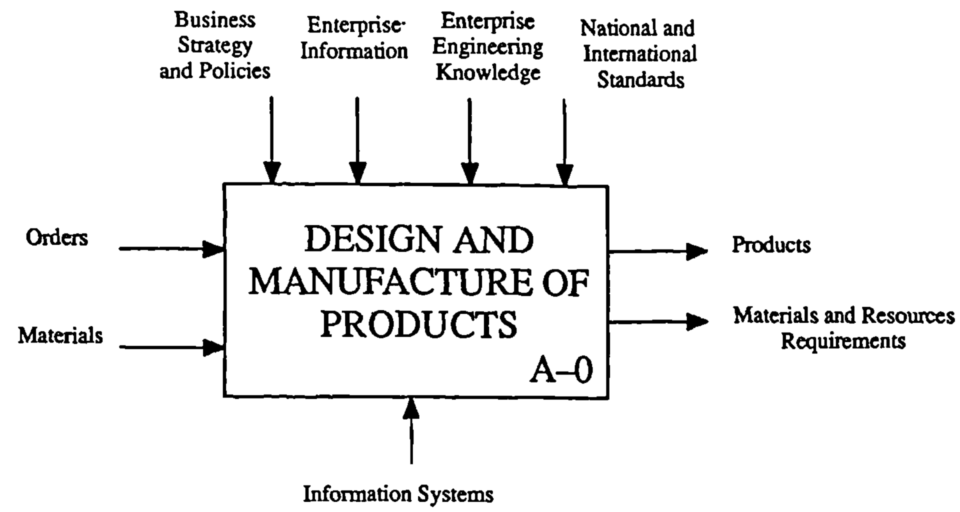
Table 2	SIMULTANEOUS ENGINEERING SYSTEMS IMPLEMENTATION CHARACTERISTICS	LUT–SE Research Group

Appendix B

IDEFO Model of Design and Manufacture of Products

This appendix presents the IDEF0 Model of the wider applicability of the Manufacturing Model.

USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 25/11/94 REV:	WORKING DRAFT X RECOMMENDED PUBLICATION	READER	DATE	CONTEXT Top

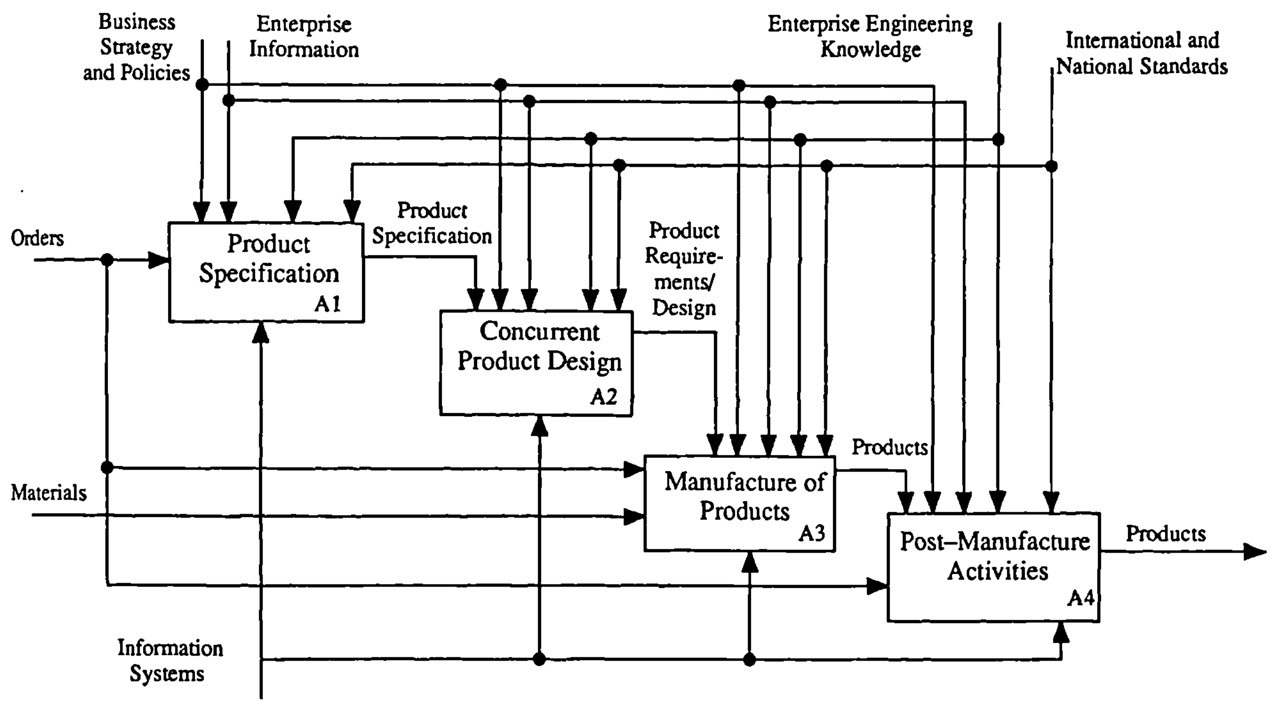


Purpose: to model the activities required to design and manufacture products using Simultaneous Engineering Principles

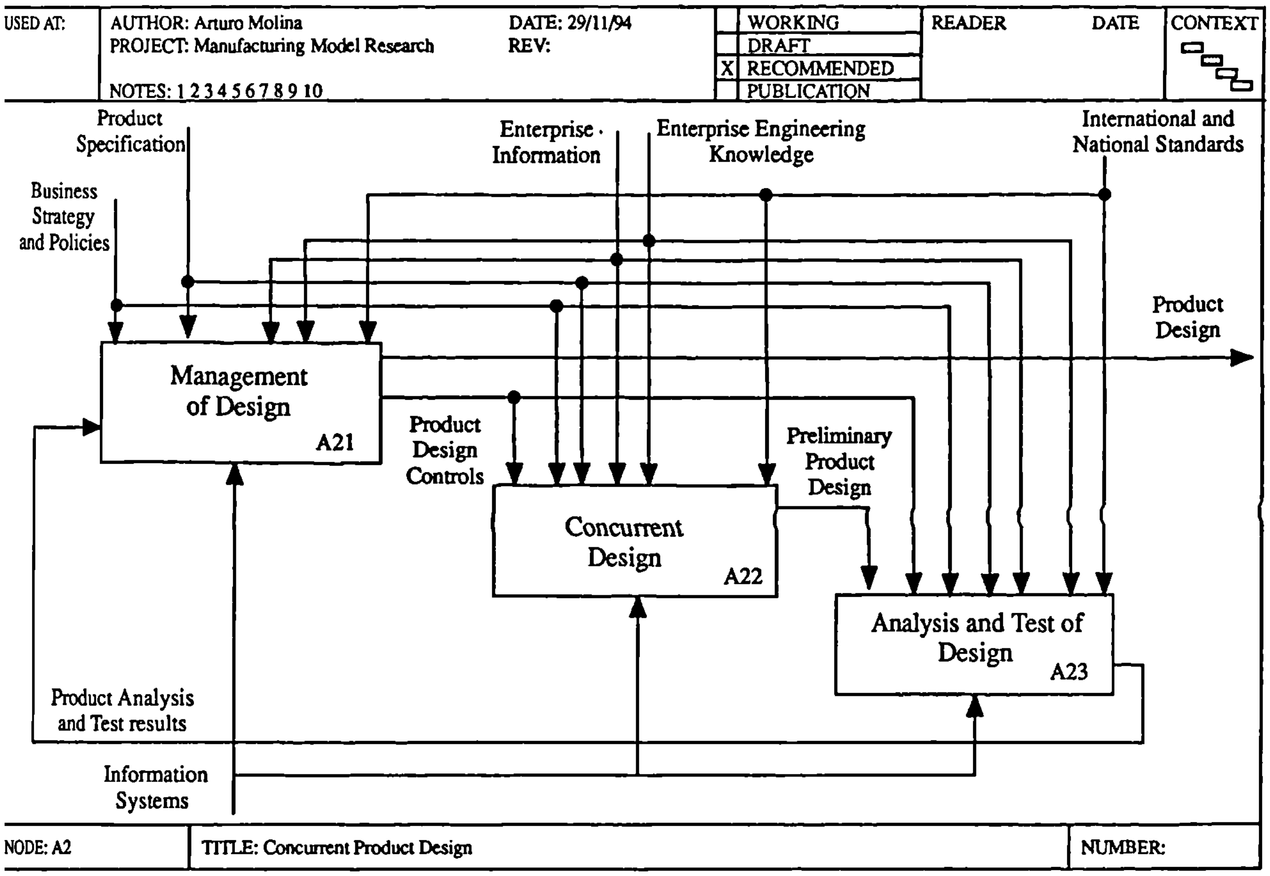
Viewpoint: a Simultaneous Engineering Team which intents to specify the product and manufacturing information requirements

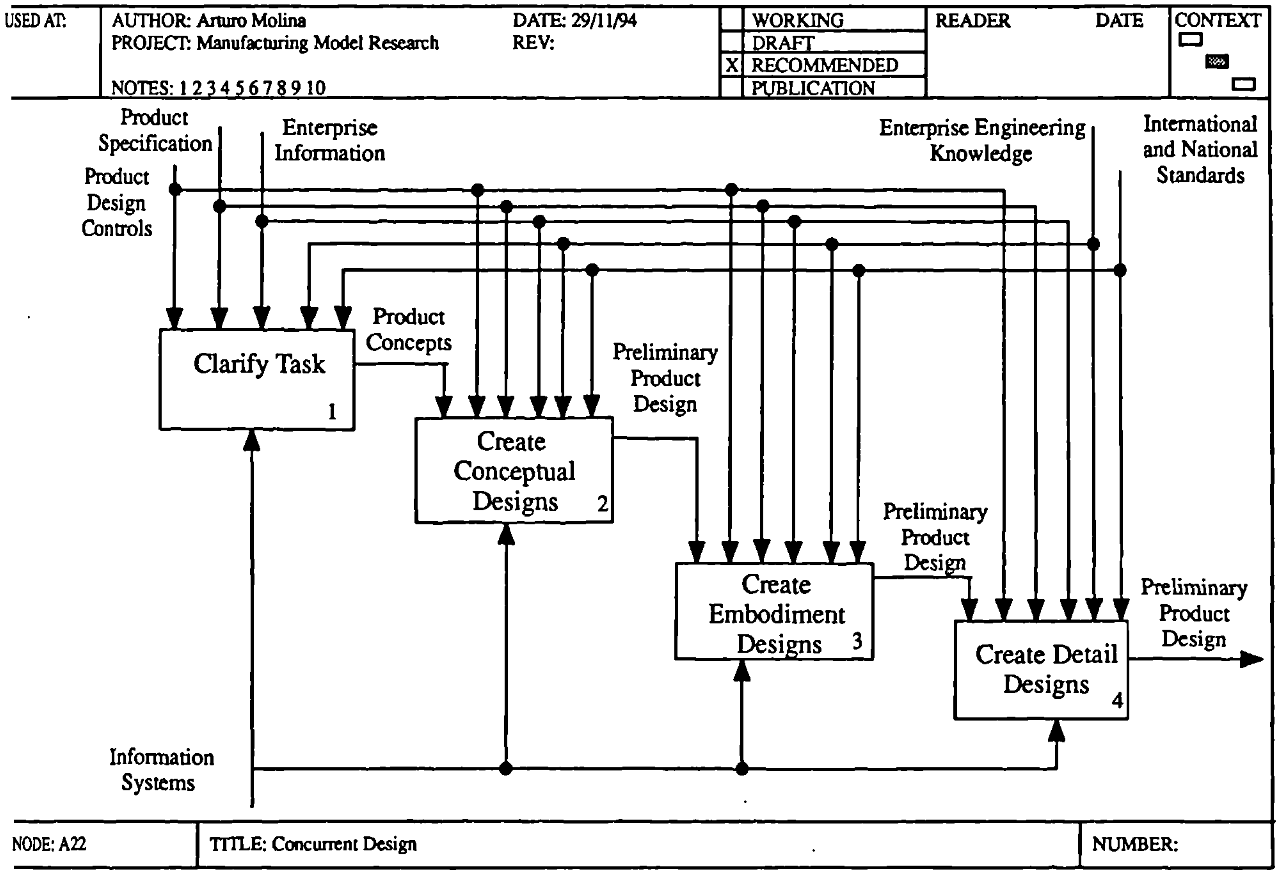
NODE: A–0	TITLE: Concurrent Product Development	NUMBER:

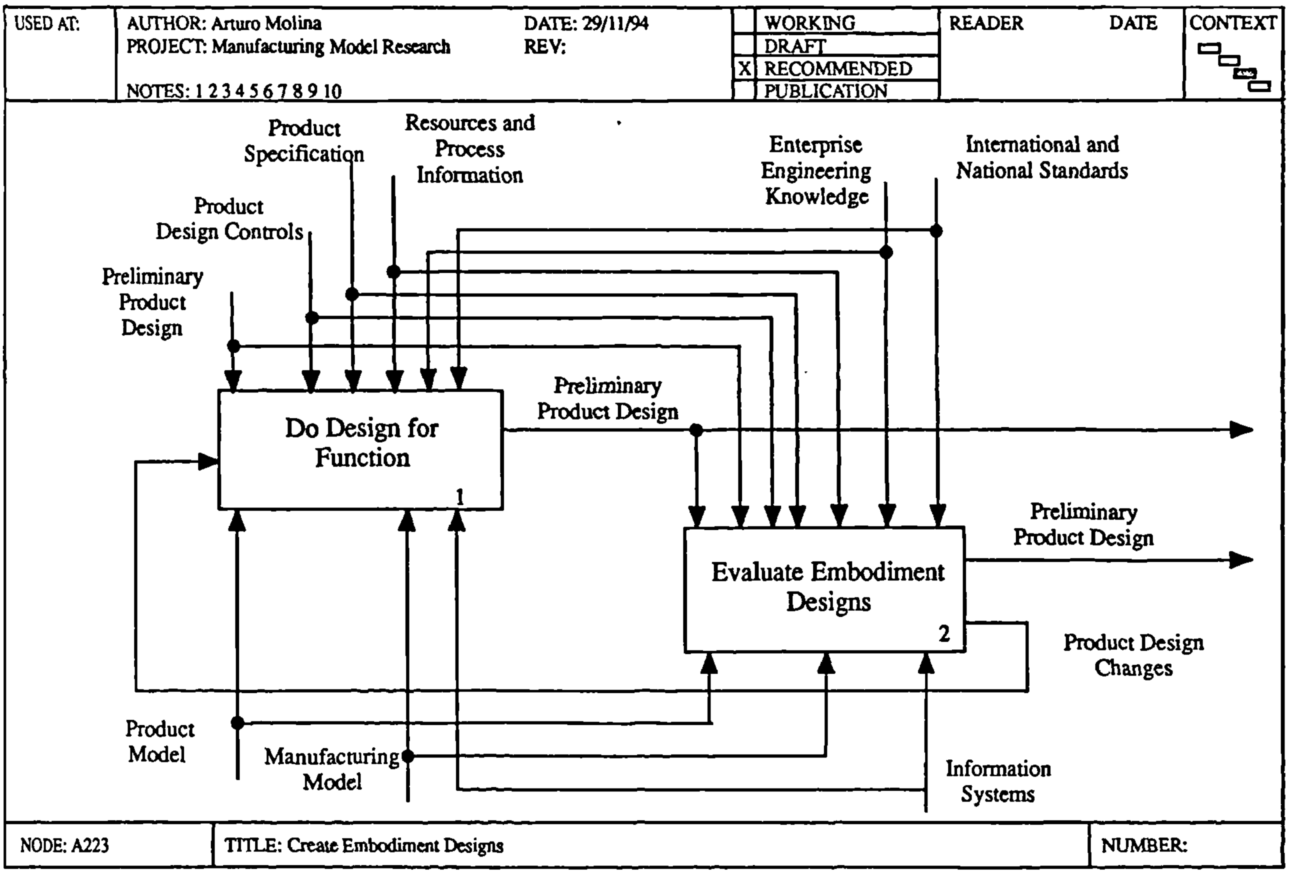
USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 29/11/94 REV:	WORKING DRAFT X RECOMMENDED PUBLICATION	READER	DATE	CONTEXT

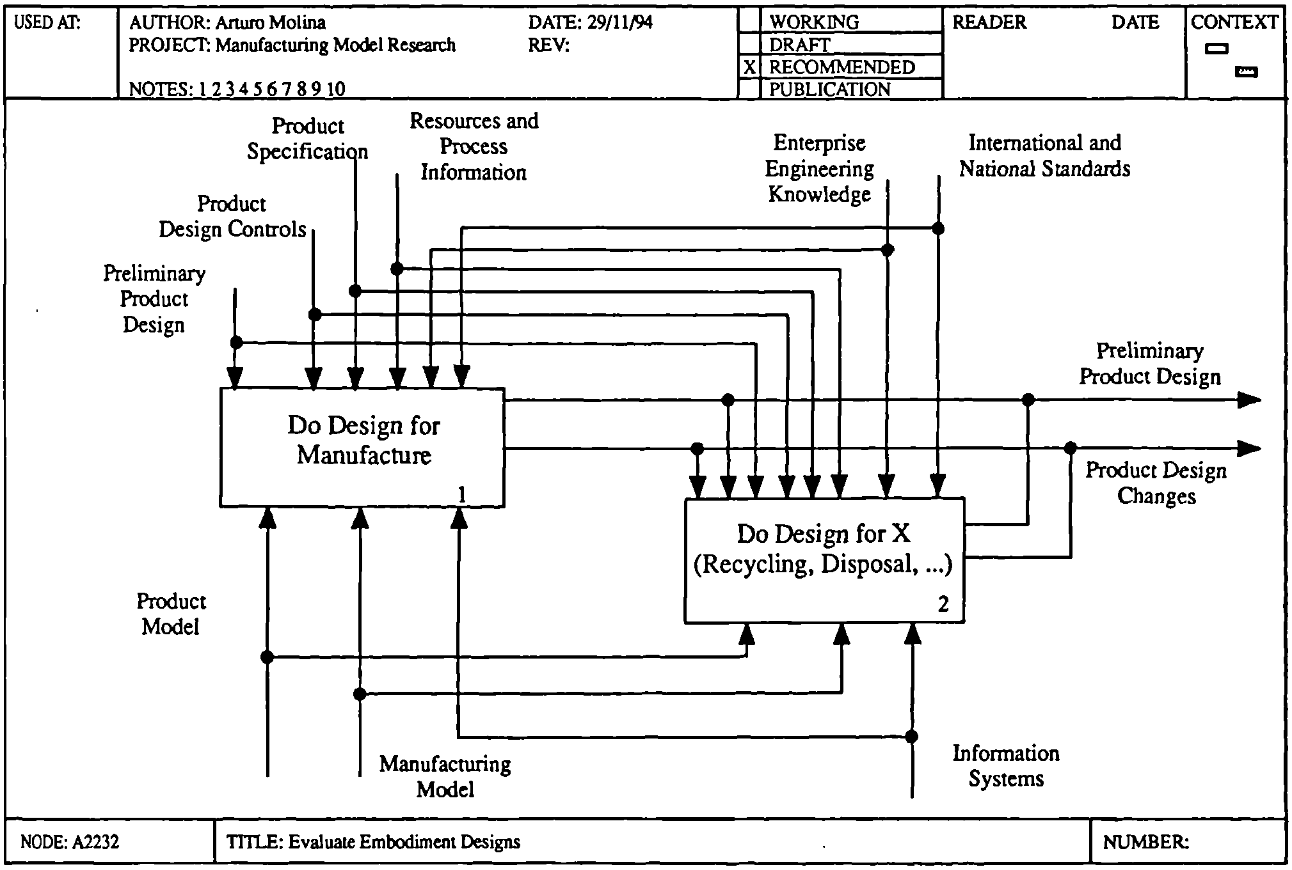


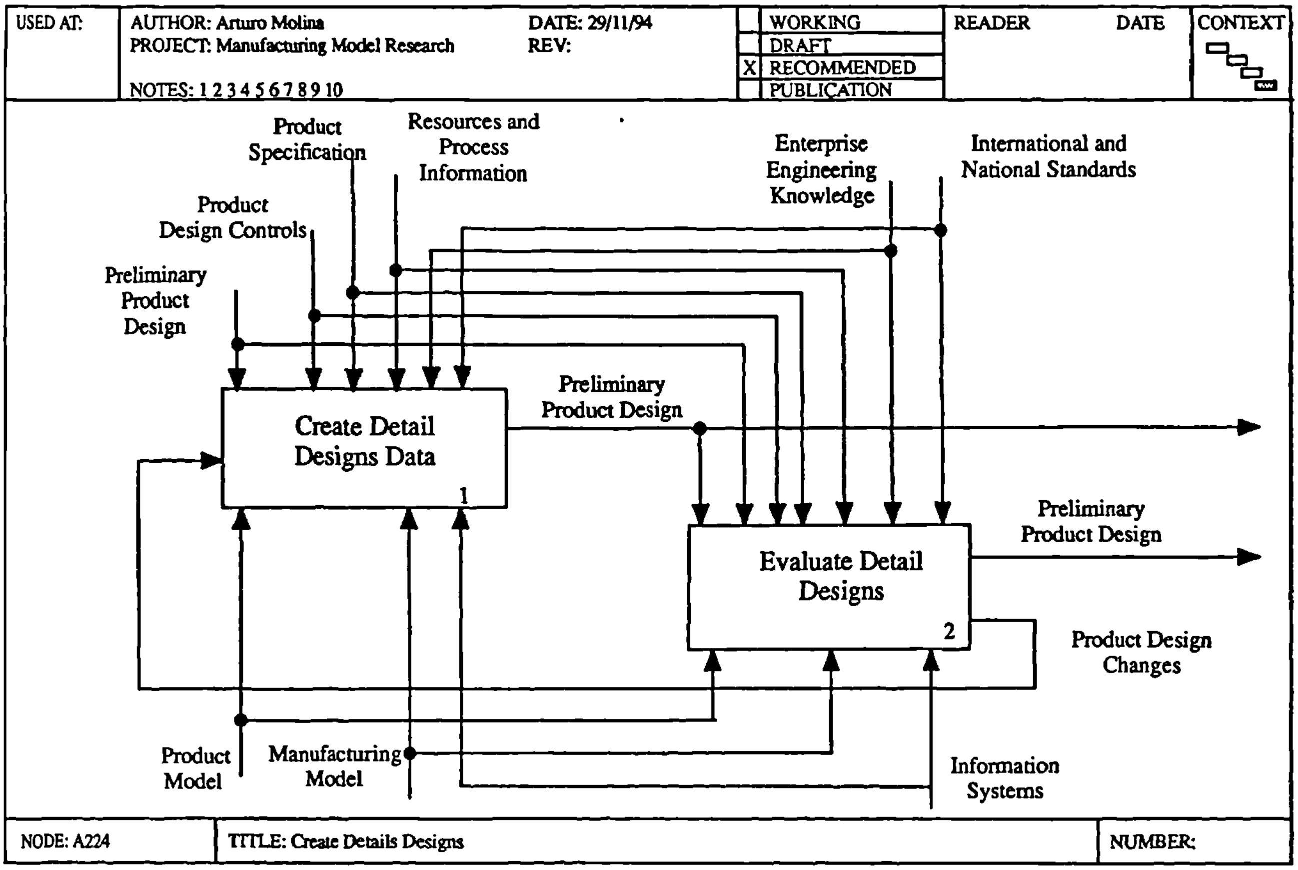
NODE: A0	TITLE: Design and Manufacture of Products	NUMBER:

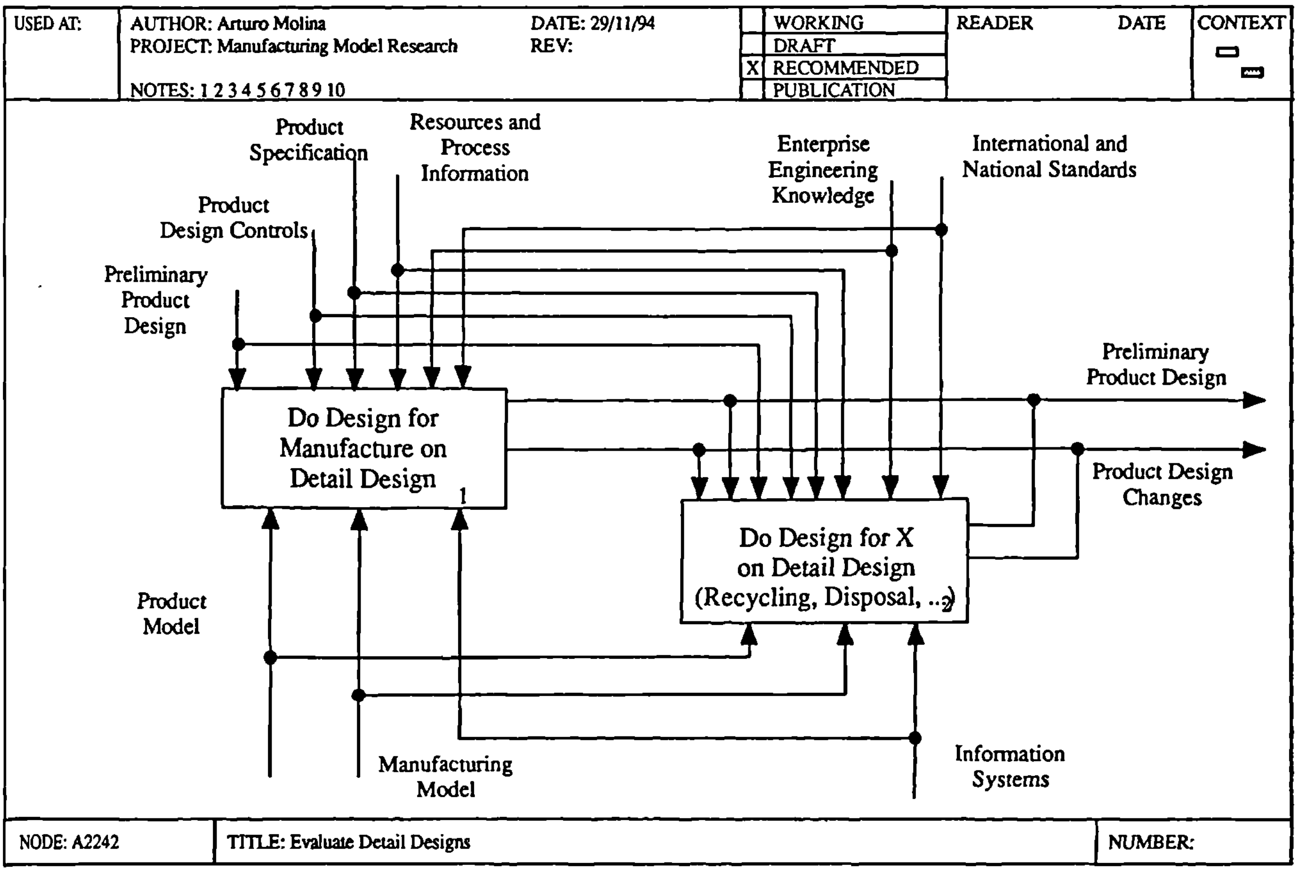

USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Business Strategy and Policies
Enterprise Information
Enterprise Engineering Knowledge
International and National Standards
Product Design
Management of Design
A21
Product Design Controls
Concurrent Design
A22
Preliminary Product Design
Analysis and Test of Design
A23
Product Analysis and Test results
Information Systems
NODE: A2
TITLE: Concurrent Product Design
NUMBER:

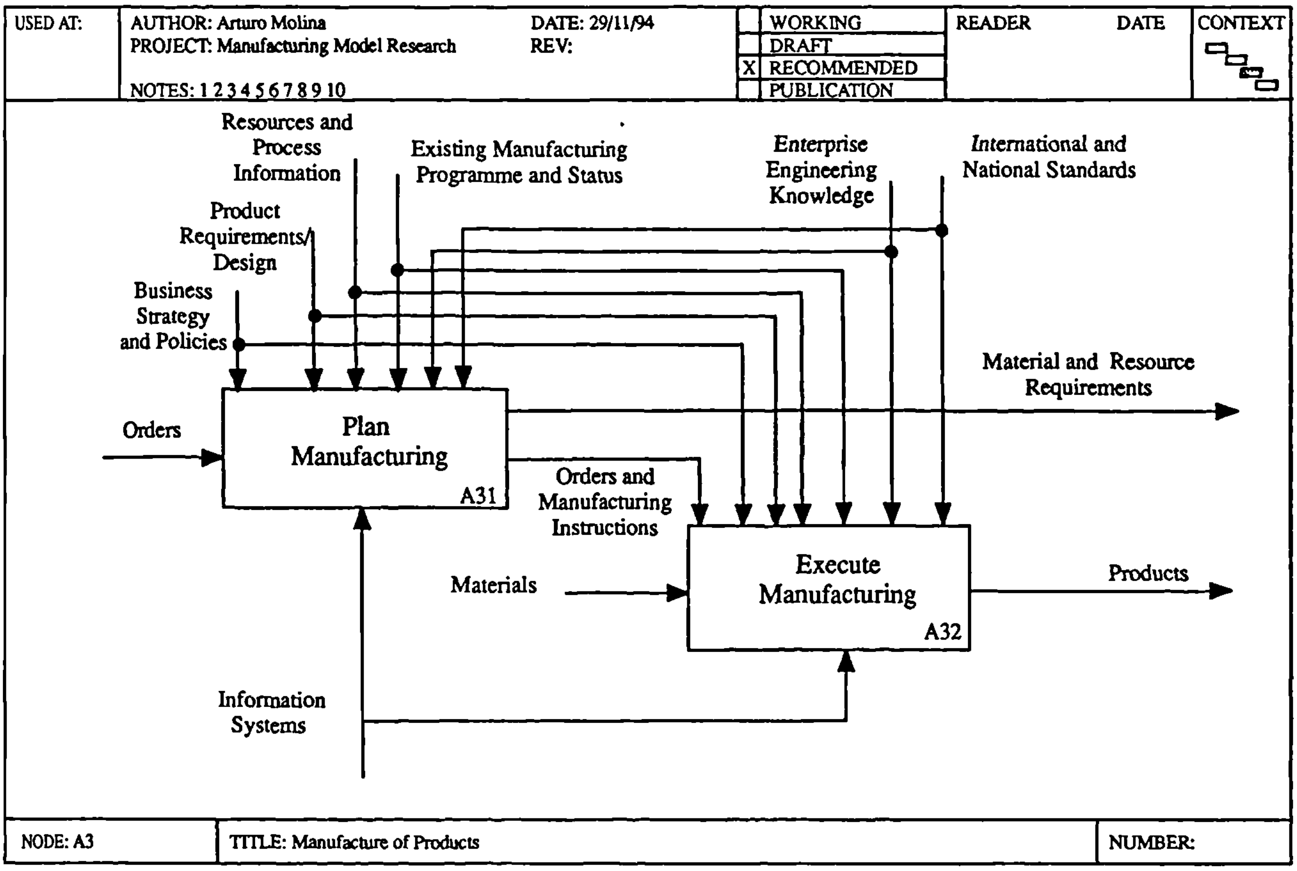

USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Product Design Controls
Enterprise Information
Enterprise Engineering Knowledge
International and National Standards
Clarify Task
1
Product Concepts
Create Conceptual Designs
2
Preliminary Product Design
Create Embodiment Designs
3
Preliminary Product Design
Create Detail Designs
4
Preliminary Product Design
Information Systems
NODE: A22
TITLE: Concurrent Design
NUMBER:

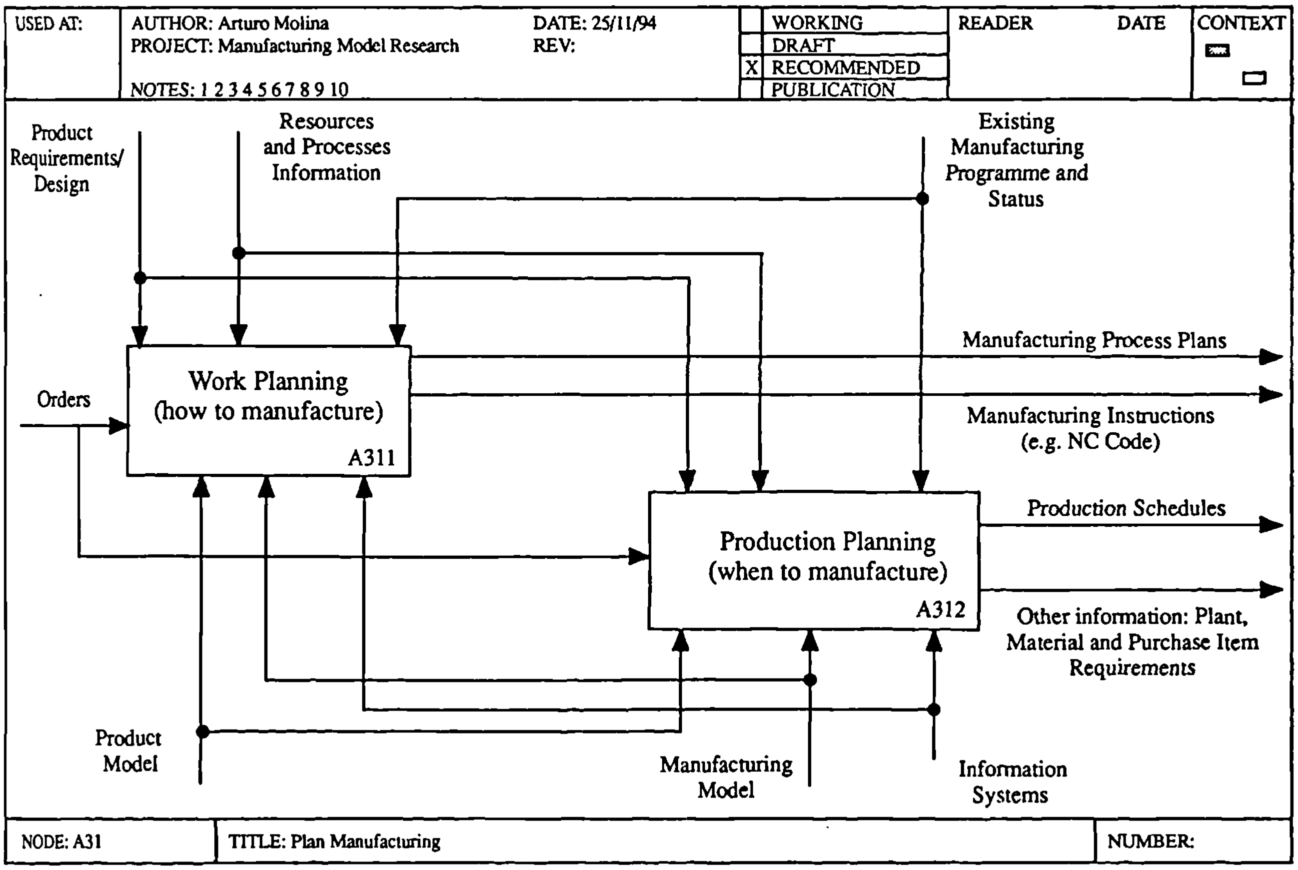

USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Resources and Process Information
Enterprise Engineering Knowledge
International and National Standards
Product Design Controls
Preliminary Product Design
Do Design for Function
1
Preliminary Product Design
Evaluate Embodiment Designs
2
Preliminary Product Design
Product Design Changes
Product Model
Manufacturing Model
Information Systems
NODE: A223
TITLE: Create Embodiment Designs
NUMBER:

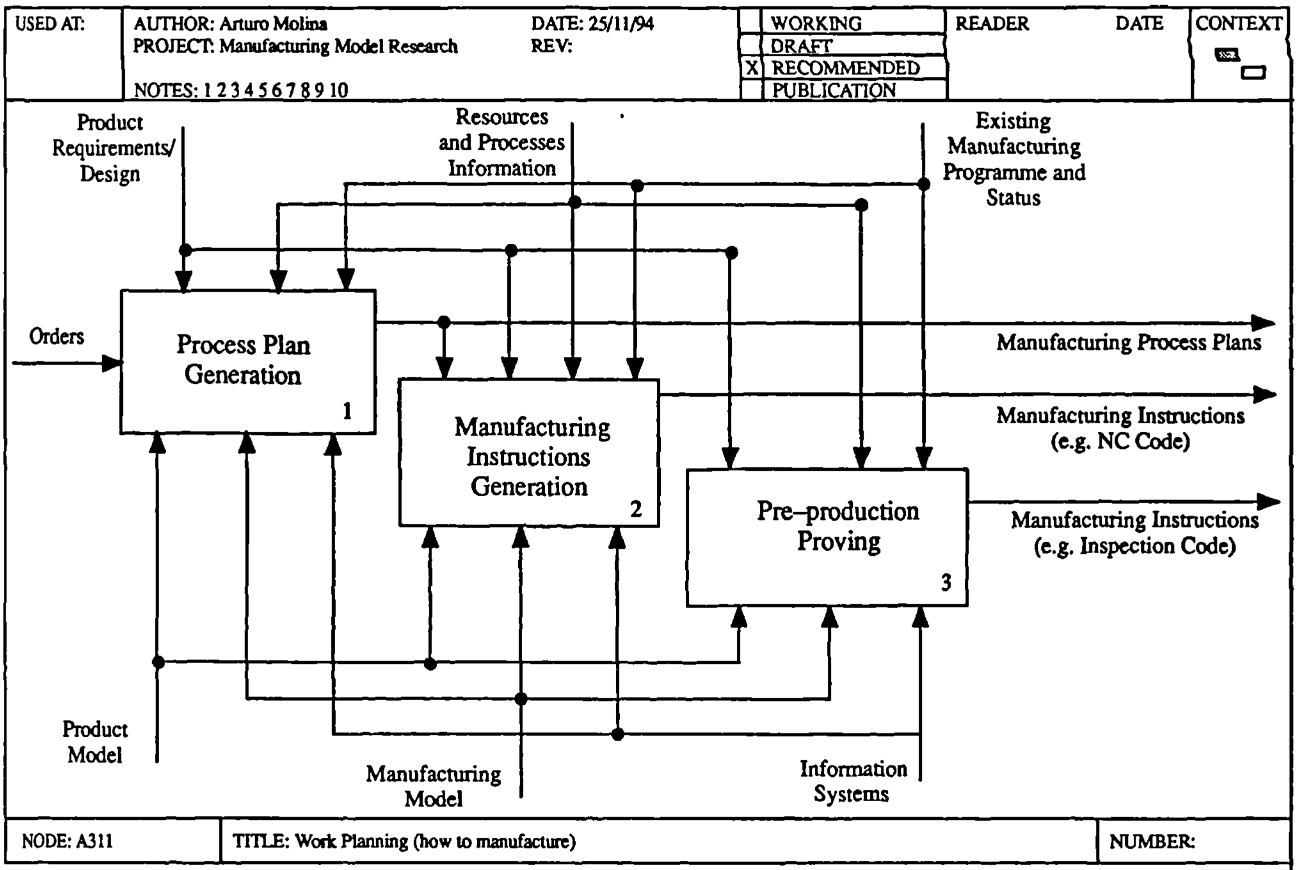

USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Resources and Process Information
Enterprise Engineering Knowledge
International and National Standards
Product Design Controls
Preliminary Product Design
Do Design for Manufacture
1
Preliminary Product Design
Product Design Changes
Do Design for X (Recycling, Disposal, ...)
2
Product Model
Manufacturing Model
Information Systems
NODE: A2232
TITLE: Evaluate Embodiment Designs
NUMBER:

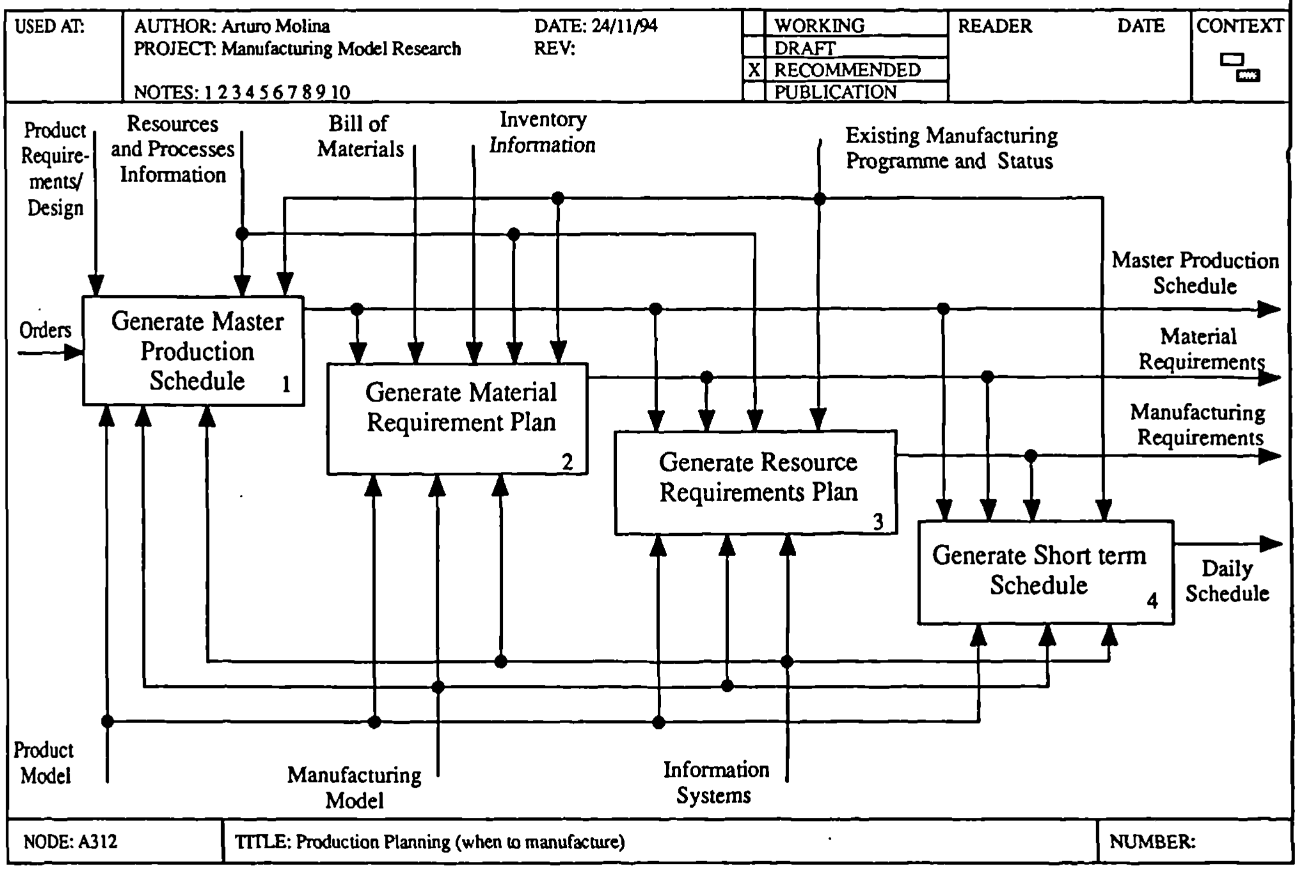

USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Resources and Process Information
Enterprise Engineering Knowledge
International and National Standards
Product Design Controls
Preliminary Product Design
Create Detail Designs Data
1
Preliminary Product Design
Evaluate Detail Designs
2
Preliminary Product Design
Product Design Changes
Product Model
Manufacturing Model
Information Systems
NODE: A224
TITLE: Create Details Designs
NUMBER:


USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Specification
Resources and Process Information
Enterprise Engineering Knowledge
International and National Standards
Product Design Controls
Preliminary Product Design
Do Design for Manufacture on Detail Design
1
Preliminary Product Design
Product Design Changes
Do Design for X on Detail Design (Recycling, Disposal, ..)
2
Product Model
Manufacturing Model
Information Systems
NODE: A2242
TITLE: Evaluate Detail Designs
NUMBER:


USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 29/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Resources and Process Information
Existing Manufacturing Programme and Status
Enterprise Engineering Knowledge
International and National Standards
Product Requirements/ Design
Business Strategy and Policies
Material and Resource Requirements
Orders
Plan Manufacturing
A31
Orders and Manufacturing Instructions
Materials
Execute Manufacturing
A32
Products
Information Systems
NODE: A3
TITLE: Manufacture of Products
NUMBER:


USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 25/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Requirements/ Design
Resources and Processes Information
Existing Manufacturing Programme and Status
Manufacturing Process Plans
Orders
Work Planning (how to manufacture)
A311
Manufacturing Instructions (e.g. NC Code)
Production Schedules
Production Planning (when to manufacture)
A312
Other information: Plant, Material and Purchase Item Requirements
Product Model
Manufacturing Model
Information Systems
NODE: A31
TITLE: Plan Manufacturing
NUMBER:


USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 25/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Requirements/ Design
Resources and Processes Information
Existing Manufacturing Programme and Status
Orders
Process Plan Generation 1
Manufacturing Instructions Generation 2
Pre–production Proving 3
Manufacturing Process Plans
Manufacturing Instructions (e.g. NC Code)
Manufacturing Instructions (e.g. Inspection Code)
Product Model
Manufacturing Model
Information Systems
NODE: A311
TITLE: Work Planning (how to manufacture)
NUMBER:


USED AT:
AUTHOR: Arturo Molina
PROJECT: Manufacturing Model Research
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 24/11/94
REV:
WORKING
DRAFT
X RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT
Product Require-ments/ Design
Resources and Processes Information
Bill of Materials
Inventory Information
Existing Manufacturing Programme and Status
Orders
Generate Master Production Schedule 1
Generate Material Requirement Plan 2
Generate Resource Requirements Plan 3
Generate Short term Schedule 4
Master Production Schedule
Material Requirements
Manufacturing Requirements
Daily Schedule
Product Model
Manufacturing Model
Information Systems
NODE: A312
TITLE: Production Planning (when to manufacture)
NUMBER:

Appendix C

Enterprise Viewpoint of the MOSES CAE Reference Model

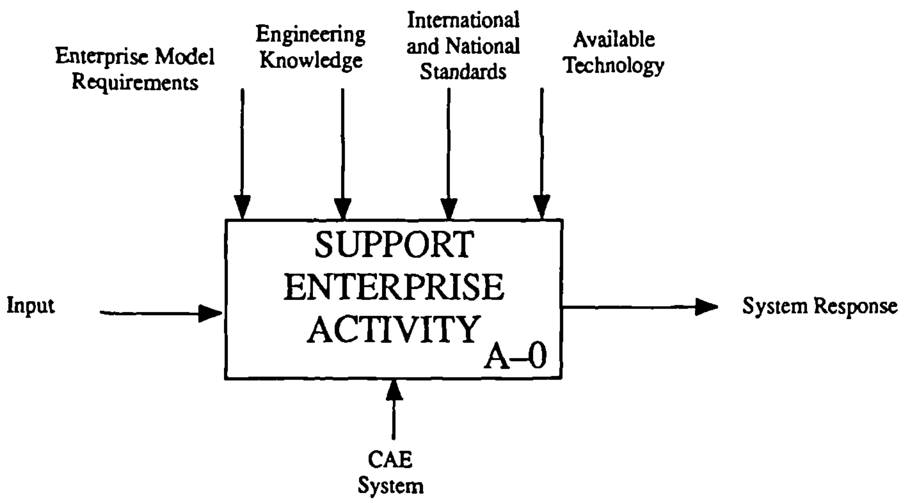
This appendix presents the Enterprise Viewpoint of the MOSES CAE Reference Model.

Activity Descriptions

A0 **- Support Enterprise Activity;**

Provide CAE support that will assist the design team for the realization of Enterprise Activities to design and and develop products quickly, cheaply and of a high quality (TIAE + AM).

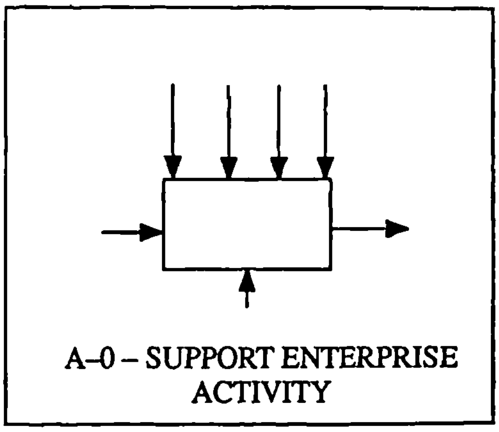
USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	WORKING DRAFT X RECOMMENDED PUBLICATION	READER	DATE



Purpose: to model the functions required in a Computer Aided Simultaneous Engineering System to satisfy the requirements of a CIMOSA Enterprise Requirements Definition Model

Viewpoint: a Simultaneous Engineering Team which intents to specify its information requirements and computer support for concurrent design of products

NODE: A-0	TITLE: Support Enterprise Activity	NUMBER:



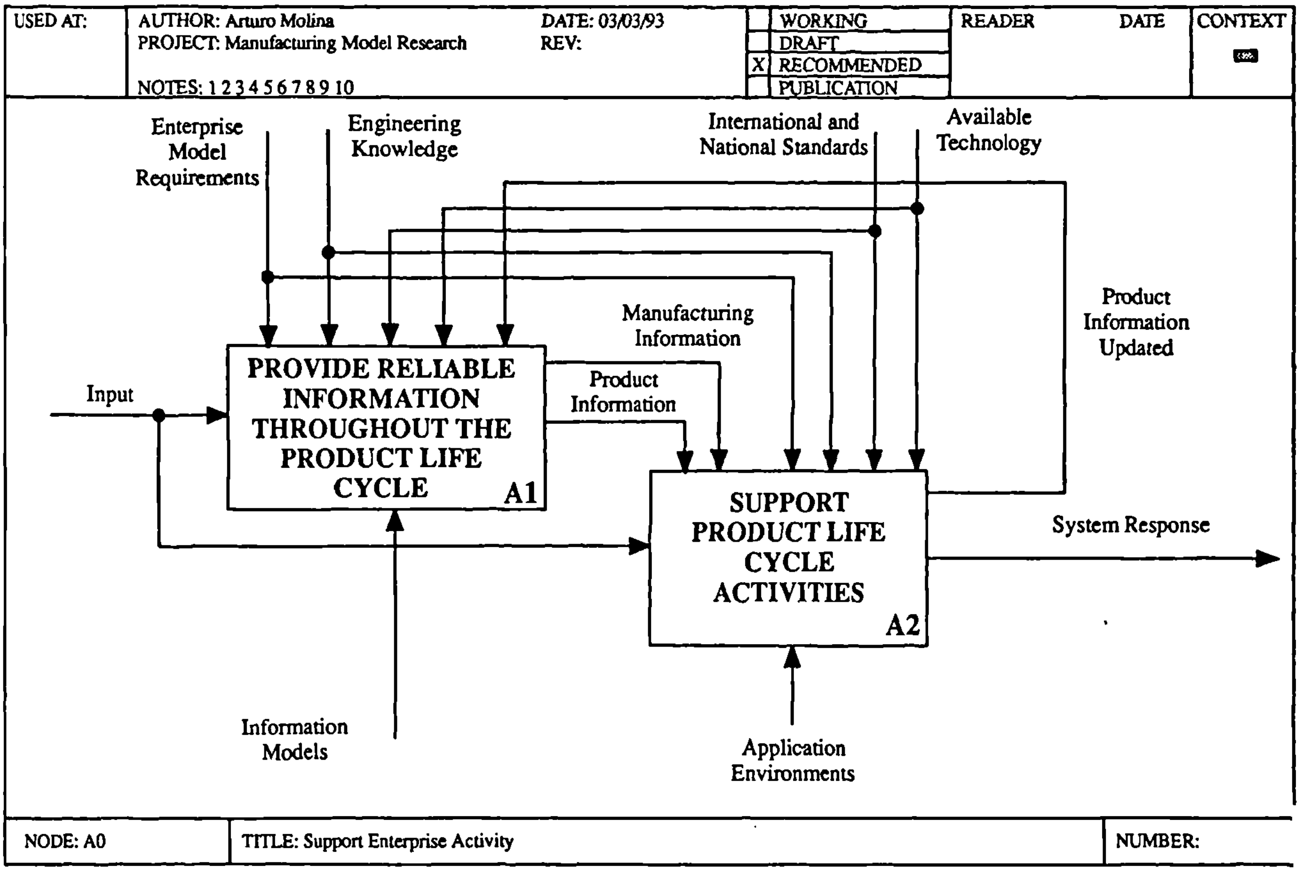
Activity Descriptions

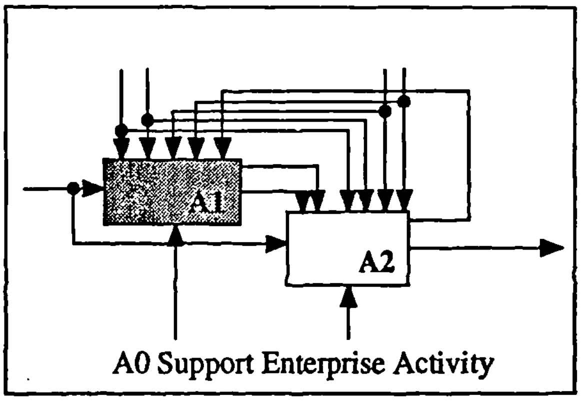
A1 – Provide Reliable Information Throughout The Product Life Cycle:

Support Enterprise Activities by providing information that is unambiguous, consistent and up to date (TIAE + AM).

A2 – Support Product Life Cycle Activities:

Support Enterprise Activities for the realization of products throughout their life cycle i.e. from conception to disposal (TIAE + AM).





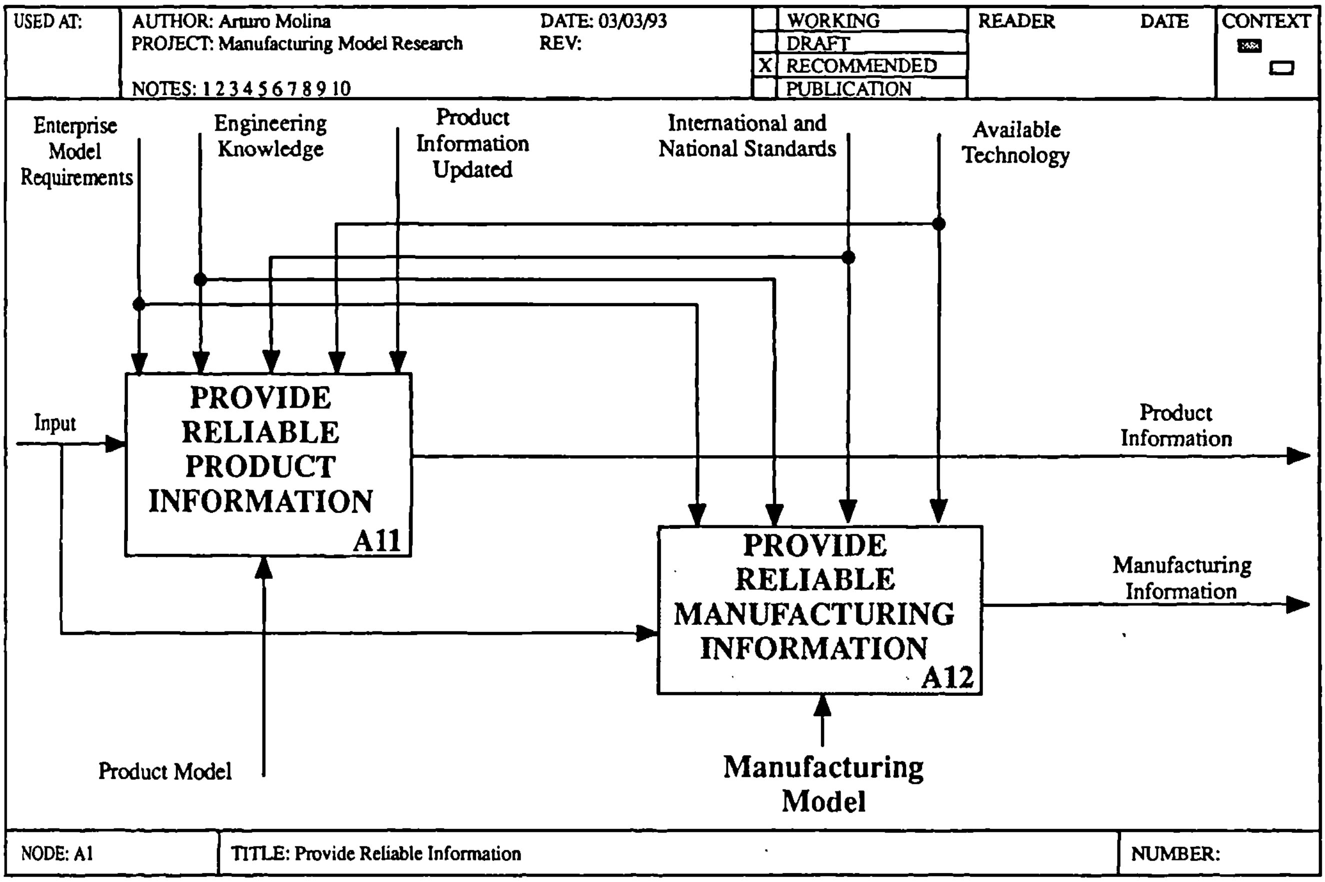
Activity Descriptions

A11 **– Provide Reliable Product Information:**

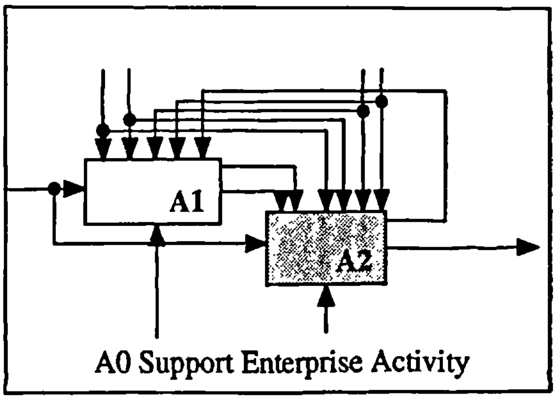
Support Enterprise Activities with information related to specific products that is unambiguous, consistent and up to date. (AM + TIAE)

A12 **– Provide Reliable Manufacturing Information:**

Support Enterprise Activities with information related to the manufacturing capabilities of the enterprise that is unambiguous, consistent and up to date. (AM)



Activity Descriptions



A21 **– Support New Product Specification:**
Provide CAE support for the production of the functional and logistical requirements of the product (TIAE)

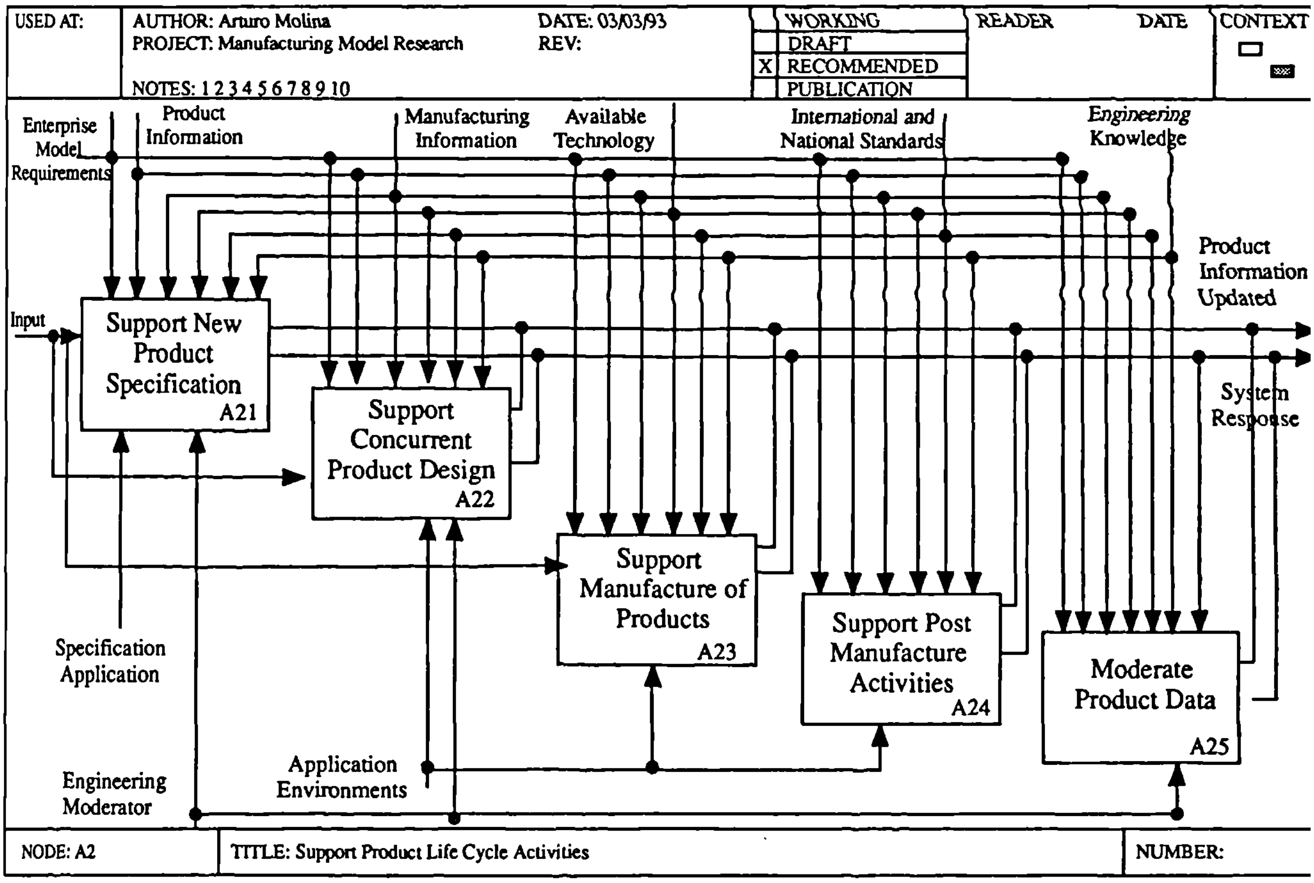
A22 **– Support Concurrent Product Design:**
Provide CAE support for the formal expression of design ideas that satisfy the functional and logistical requirements of the product using simultaneous engineering principles (AM + TIAE)

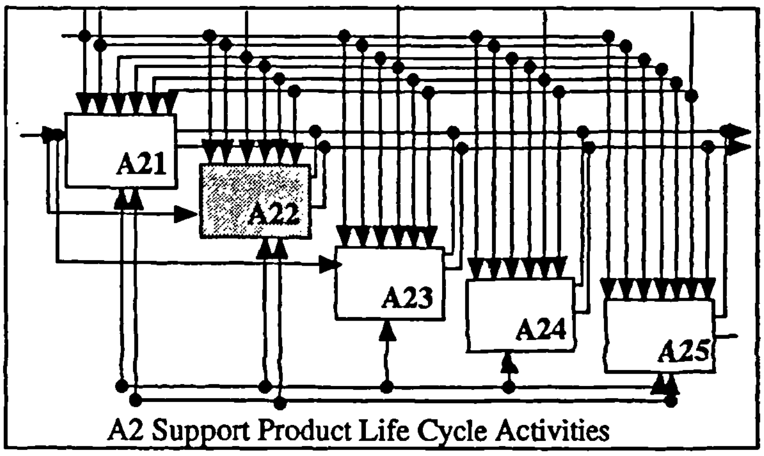
A23 **– Support Manufacture of Products:**
Provide CAE support for the conversion of a product into a physical object (TIAE)

A24 **– Support Post Manufacture Activities:**
Provide CAE support for all activities immediately after the manufacture of the product until its final disposal (TIAE)

A25 **– Moderate Product Data:**

Examine the product model for conflicts within the design. Conflict exists if it is possible to predict that problems will occur at a future stage in the product life cycle, by using the information within the information models. Alternatively, conflict exists if any application could find a significant flaw within the design, or recommend a significant improvement to the design (JAH).





Activity Descriptions

A221 **– Support Management of Design:**

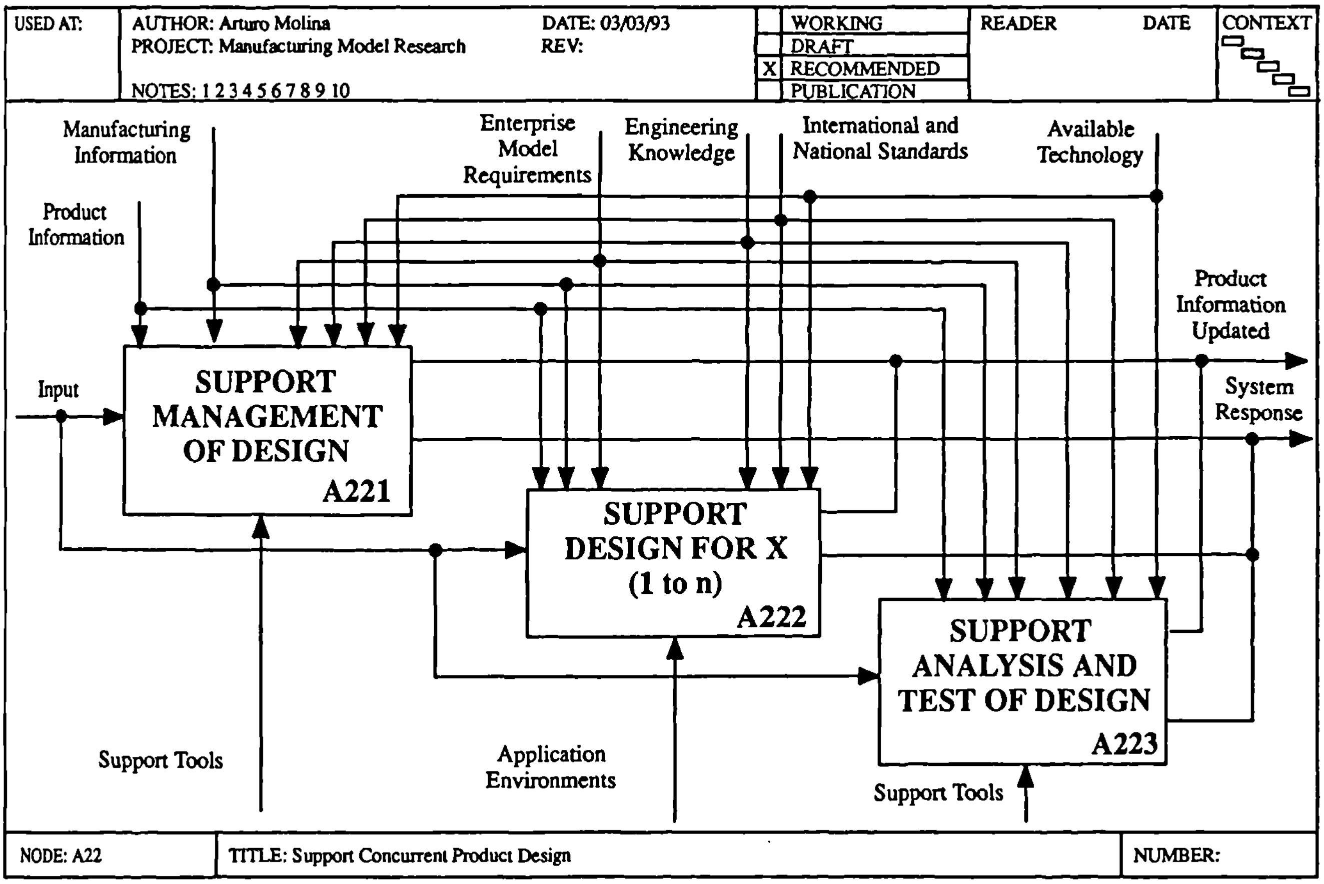
Provide CAE support for the creation of schedules, budgets and staffing requirements for the design of process (TIAE)

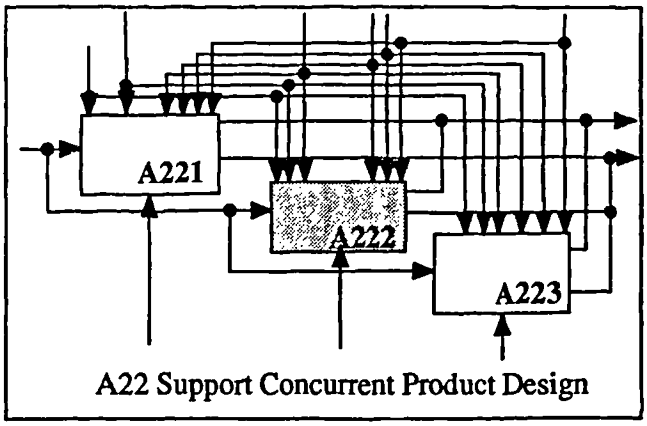
A222 **– Support Design for X (1 to n):**

Provide CAE support for a range of design activities that consider the influence of particular life cycle activities on the form and function of an evolving design (TIAE)

A222 **– Support Analysis and Test of Design:**

Provide CAE support for the creation of data during the design process that is to be used to evaluated the design, and for the process of assessing a product's design against its requirements (TIAE)





Activity Descriptions

A2221 **– Support Design for Function:**

Provide CAE support for the design activity that considers the influence of functional requirements on the form and content of the product design (TIAE)

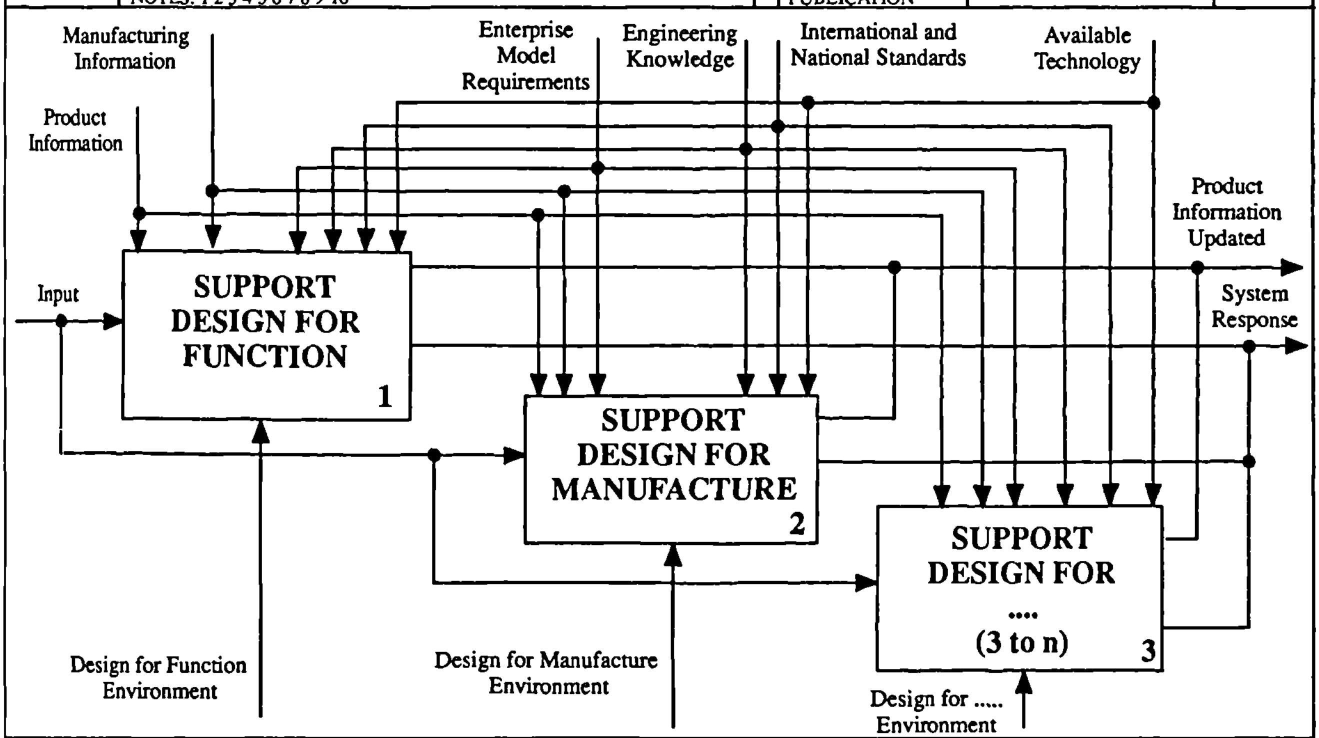
A2222 **– Support Design for Manufacture:**

Provide CAE support for the design activity that considers how the way in which a product will be manufactured influences product design (TIAE)

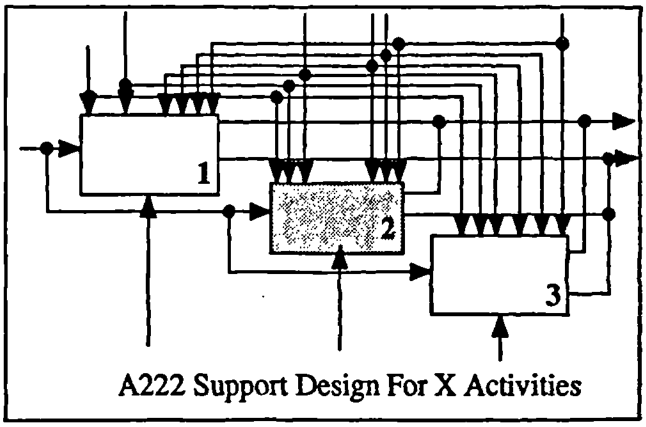
A2223 **– Support Design for .... (3 to n):**

Provide CAE support for a range of design activities that consider the influence of life cycle activities on the form and function of an evolving design. These design activities exclude Design for Function and Design for Manufacture (TIAE).

USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	WORKING DRAFT X RECOMMENDED PUBLICATION	READER	DATE	CONTEXT



NODE: A222	TITLE: Support Design For X Activities	NUMBER:



Activity Descriptions

1 **– Support Design for Assembly:**

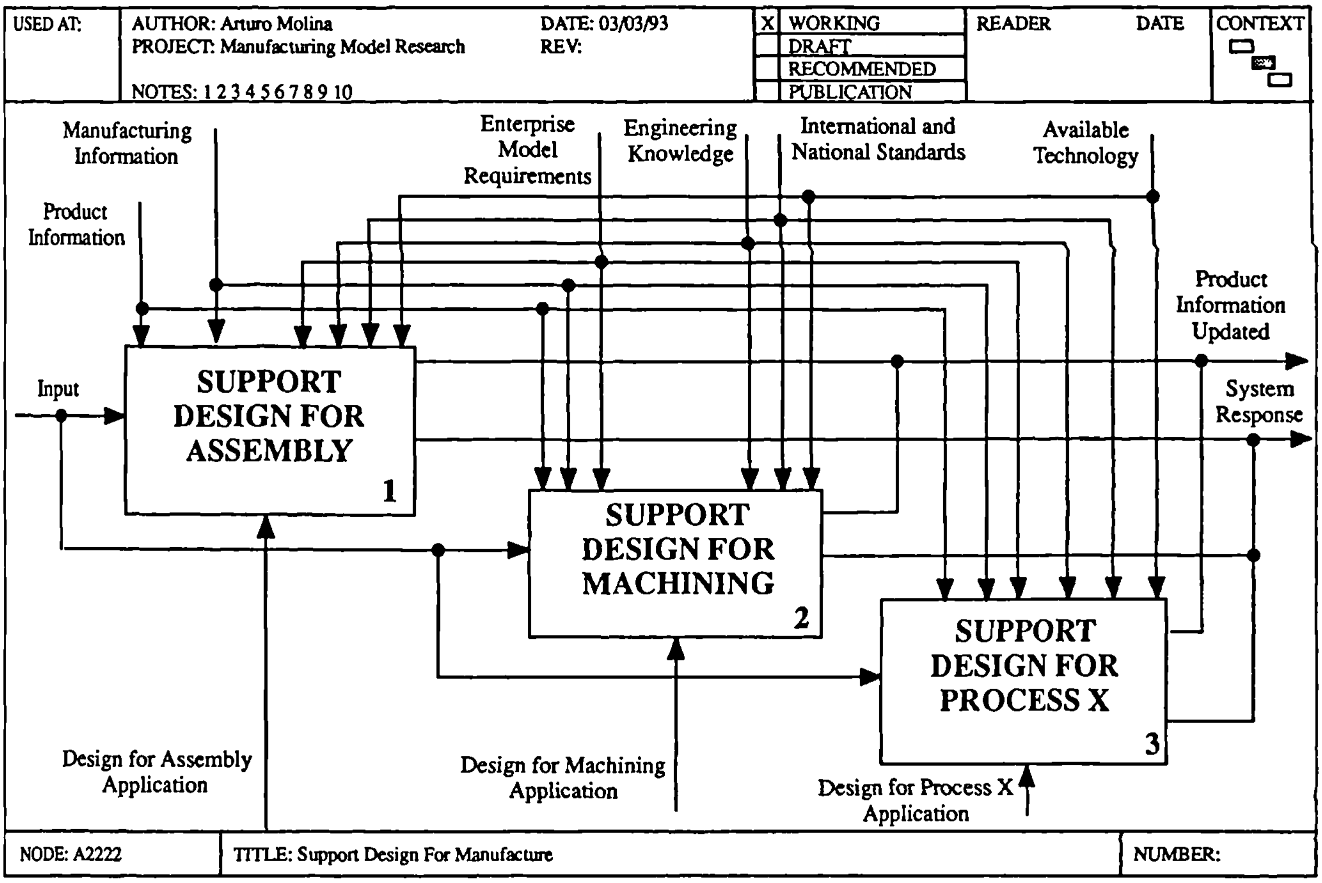
Provide CAE support for the design activity that *considers how the way in which a product will be assembled* influences product design (AM).

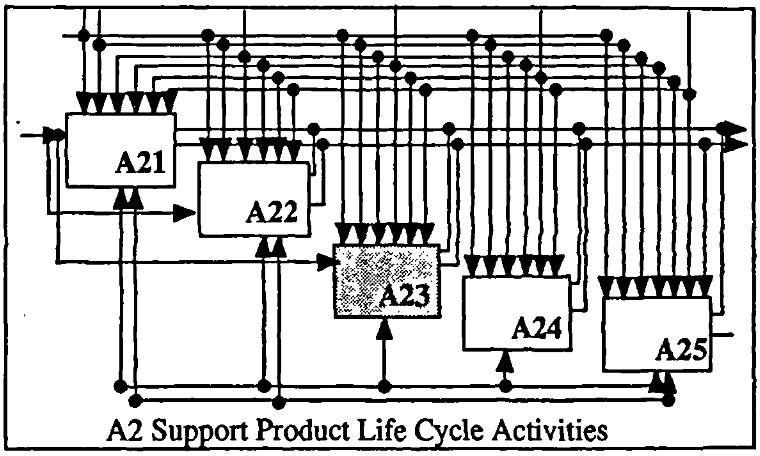
2 **– Support Design for Machining:**

Provide CAE support for the design activity that considers how the way in which a product will be machined influences product design (AM)

3 **– Support Design for Process X:**

Provide CAE support for the design activity that considers how the way in which a product manufactured by a process X influences product design (AM)





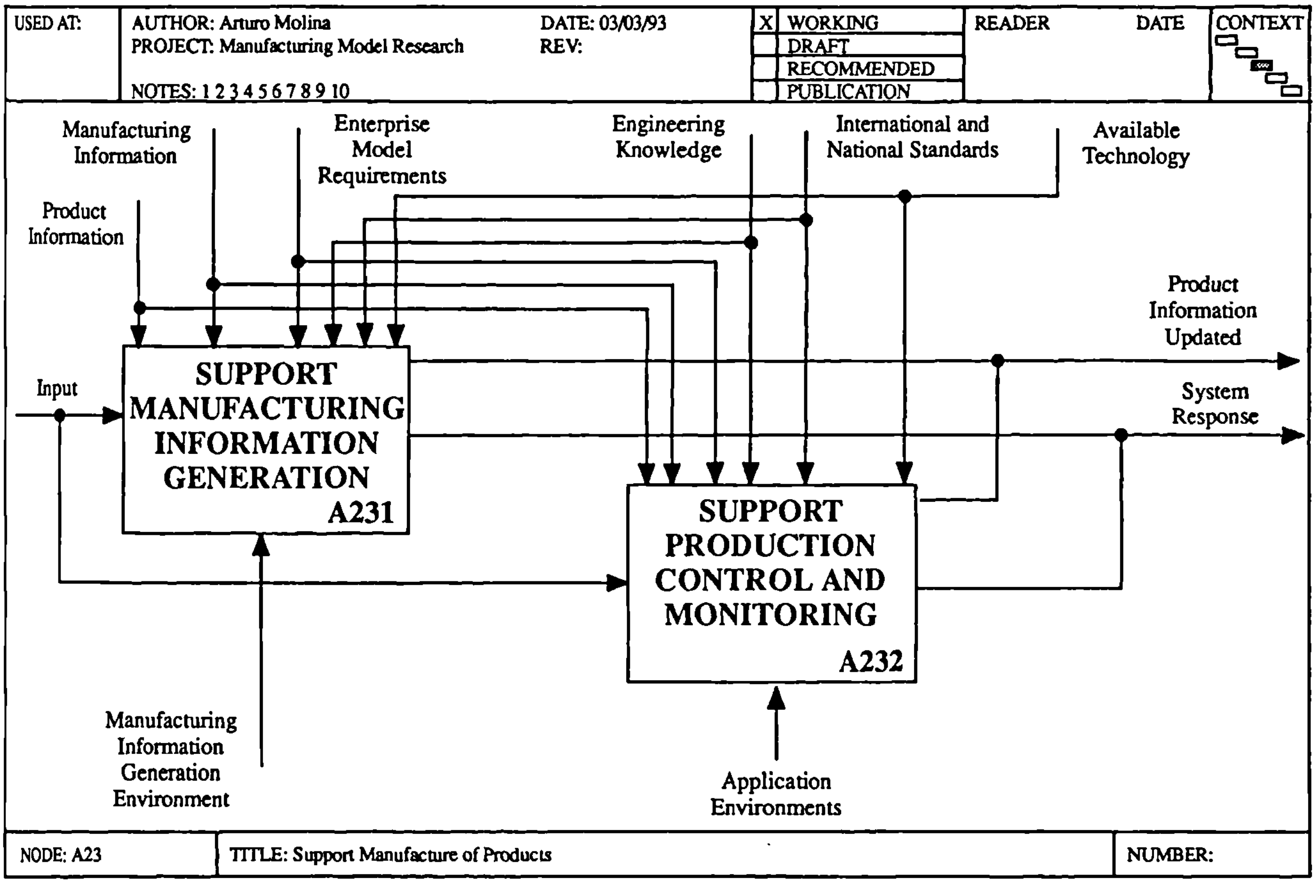
Activity Descriptions

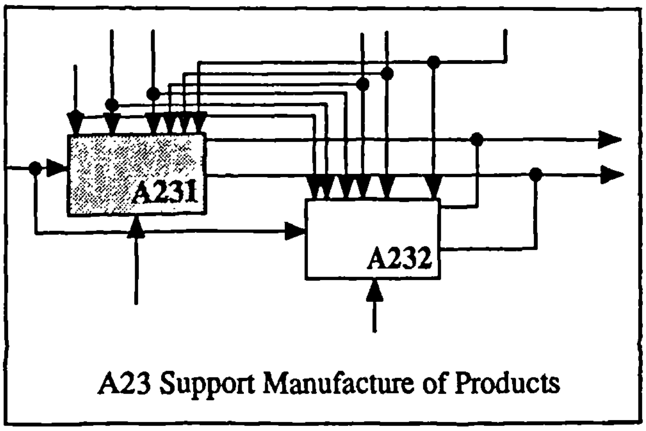
A231 **– Support Manufacturing Information Generation:**

Provide CAE support for the activities that generates manufacturing information to manufacture a product (AM).

A232 **– Support Production Control and Monitoring:**

Provide CAE support for the activities that controls and monitors the real time production of a product (AM)





Activity Descriptions

1 **– Support Process Planning Generation:**

Provide CAE support for the activities that generates process plans to manufacture a product (AM).

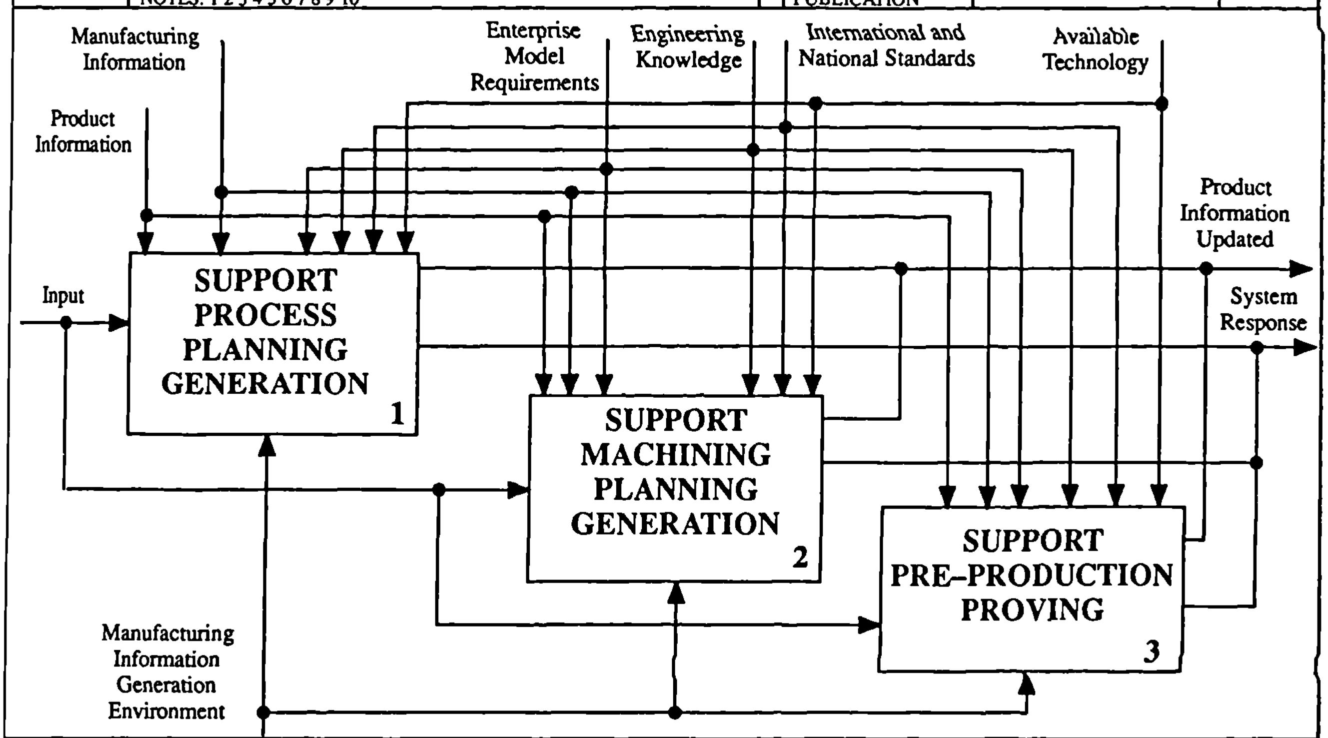
2 **– Support Machining Planning Generation:**

Provide CAE support for the activities that generates machining plans (NC–Code) to manufacture a product (AM).

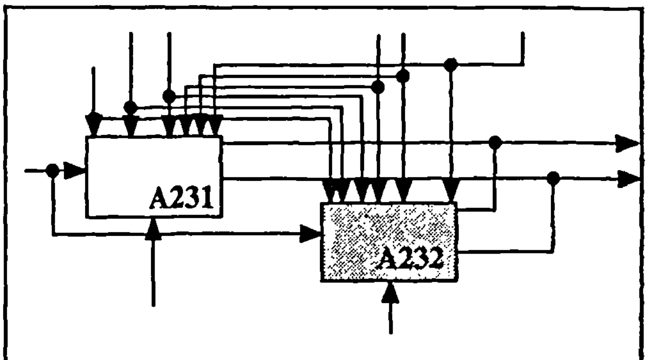
3 **– Support Pre–Production Proving:**

Provide CAE support for the activities that generates code for the pre–production proving of a product (AM).

USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	X WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT



NODE: A231	TITLE: *Support Manufacturing Information Generation*	*NUMBER:*



A23 Support Manufacture of Products

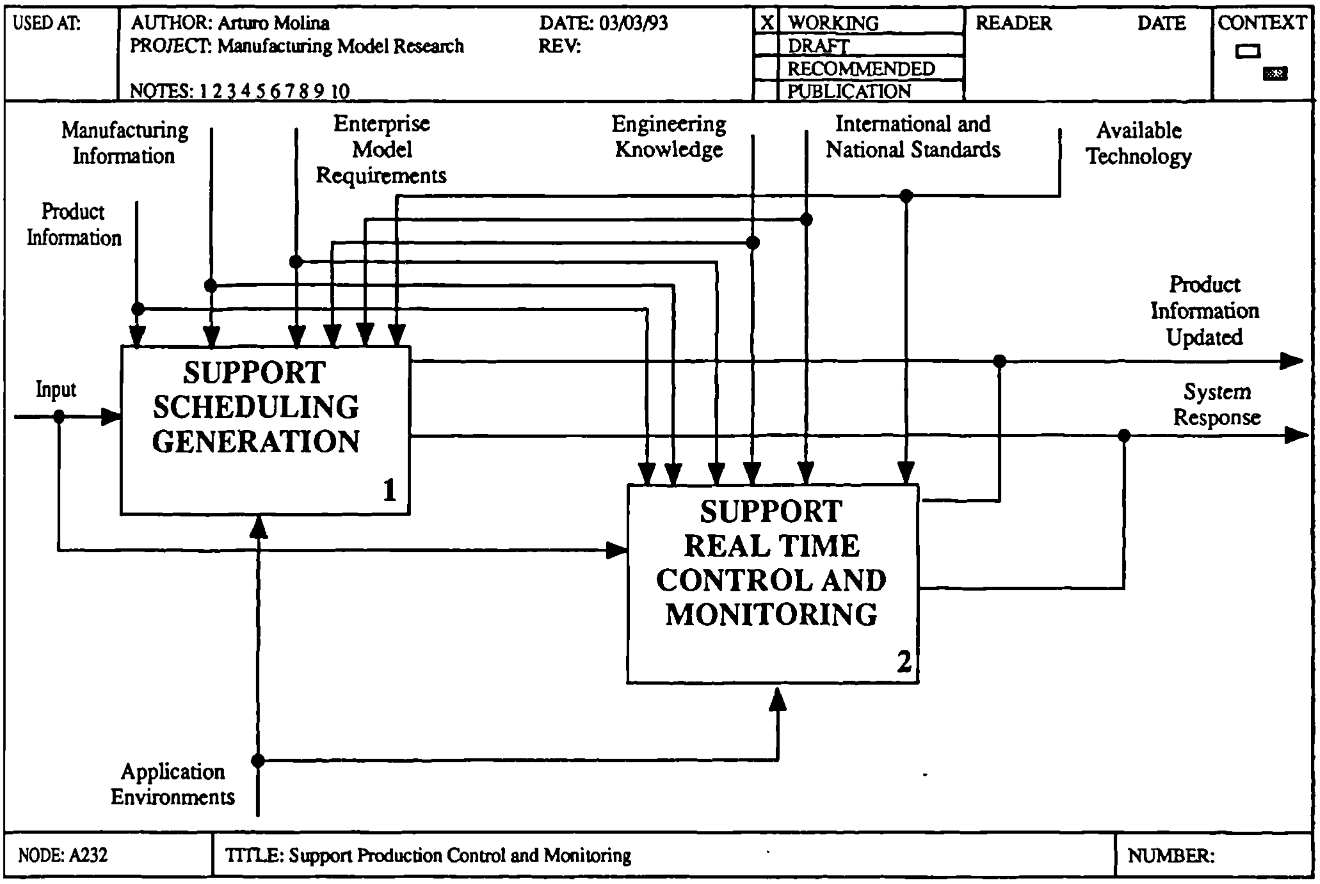
Activity Descriptions

A2321 **– Support Schedule Generation:**

Provide CAE support for the activities that generates schedules for the production of a product (AM).

A2322 **– Support Production Control and Monitoring:**

Provide CAE support for the activities that enable the production of a product (AM).



Glossary of terms of the IDEF0 Model

Application Environment
A set of data–driven applications and software tools used to provide CAE support for an activity. Different application environment can exist i.e. Design for Function Environment, Design for Manufacture Environment, etc.

Available Technology
The equipment, processes and computing facilities that the company has knowledge of, and access to, either directly or through a sub–contractor.

Design for Assembly Application
The software tools used to provide CAE support for the Design for Assembly Activity.

Design for Function Environment
A set of applications and software tools used to provide CAE support for the Design for Function Activity.

Design for Machining Application
The software tools used to provide CAE support for the Design for Machining Activity.

Design for Manufacture Environment
A set of applications and software tools used to provide CAE support for the Design for Manufacture Activity

Design for Process X Application
The software tools used to provide CAE support for the Design for Process X Activity.

Design for .... Environment
A set of applications and software tools used to provide CAE support for the Design for .... Activity

Enterprise Model Requirements:
Requirements defined in the CIMOSA Enterprise Requirements Definition Model which are required to be satisfy by the CAE System functions.

Engineering Knowledge
Knowledge gained about products, processes and business.

Engineering Moderator
A CAE system element with the function of supporting and enforcing concurrency

Information Model
A model which represent information of some universe of discourse

International and National Standards
Standards that provide guide-lines at either national or international level relating to either products, processes or management.

Input,
Inquiry
User Input
A user query input to the CAE system or one of its elements

Manufacturing Information
Information related to Enterprise resources, processes and strategies

Manufacturing Information Generation Environment
A set of applications and software tools used to provide CAE support for the Manufacturing Information Generation Activity.

Manufacturing Model
A CAE system element which has been defined as : "An information model, which identifies, represents and captures the data, information and knowledge, that describes the manufacturing resources, processes, and strategies of a particular enterprise. This enables the provision of the necessary manufacturing information for the support of the manufacturing decision making in concurrent design of products".

Product Information
Information related to the life cycle of a product.

Product Model
A CAE system element which captures and represent the Product *information* generated throughout its life cycle.

Product Information Updated
Product information that has been modified as a result of being processes by some element of the CAE system

Response
System Response
A response to a query input from the CAE system. The response can be different depending on which element of the CAE system has been trigger. For example, a system response of the Manufacturing Model is Manufacturing Information

Specification Application
The software tools used to provide CAE support for the product specification activity.

Support Tools
The methodologies, systems and software tools used to provide CAE support for an activity.

Appendix D

Enterprise Viewpoint of the Manufacturing Model

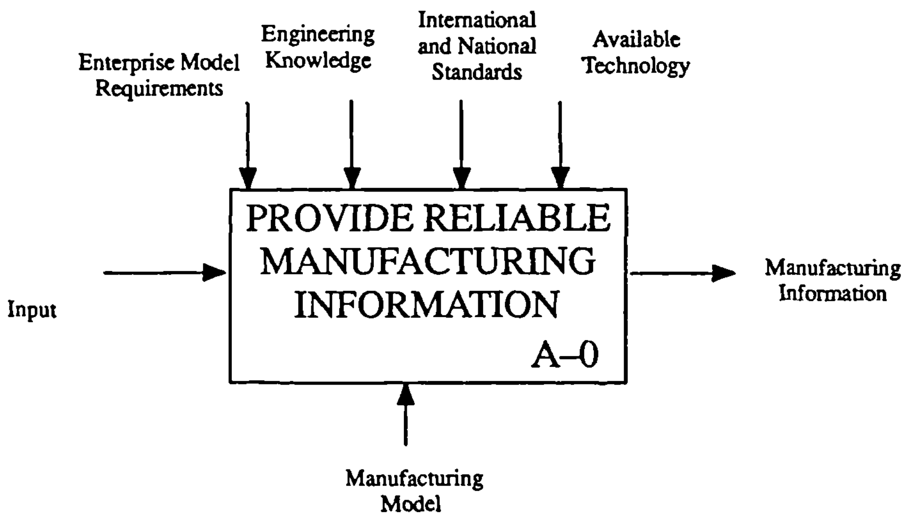
This appendix presents the Enterprise Viewpoint of the Manufacturing Model, which is related to the Enterprise Viewpoint of the MOSES CAE Reference Model.

Activity Descriptions

A0 **– Provide Reliable Manufacturing Information:**

Support Enterprise Activities and CAE System elements by providing information related to the manufacturing capabilities of the enterprise that is unambiguous, consistent and up to date (AM)

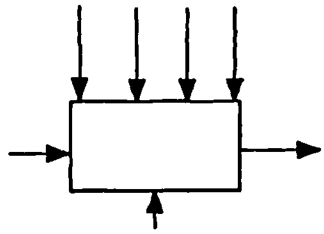
USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	 X 	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT Top



Purpose: to model the functions required in the Manufacturing Model to satisfy the manufacturing information requirements of the CIMOSA Enterprise Requirements Definition Model and CAE System

Viewpoint: a Simultaneous Engineering Team which intents to specify its manufacturing information requirements

NODE: A–0	TITLE: Provide Reliable Manufacturing Information	NUMBER:



A–0 – PROVIDE RELIABLE MANUFACTURING INFORMATION

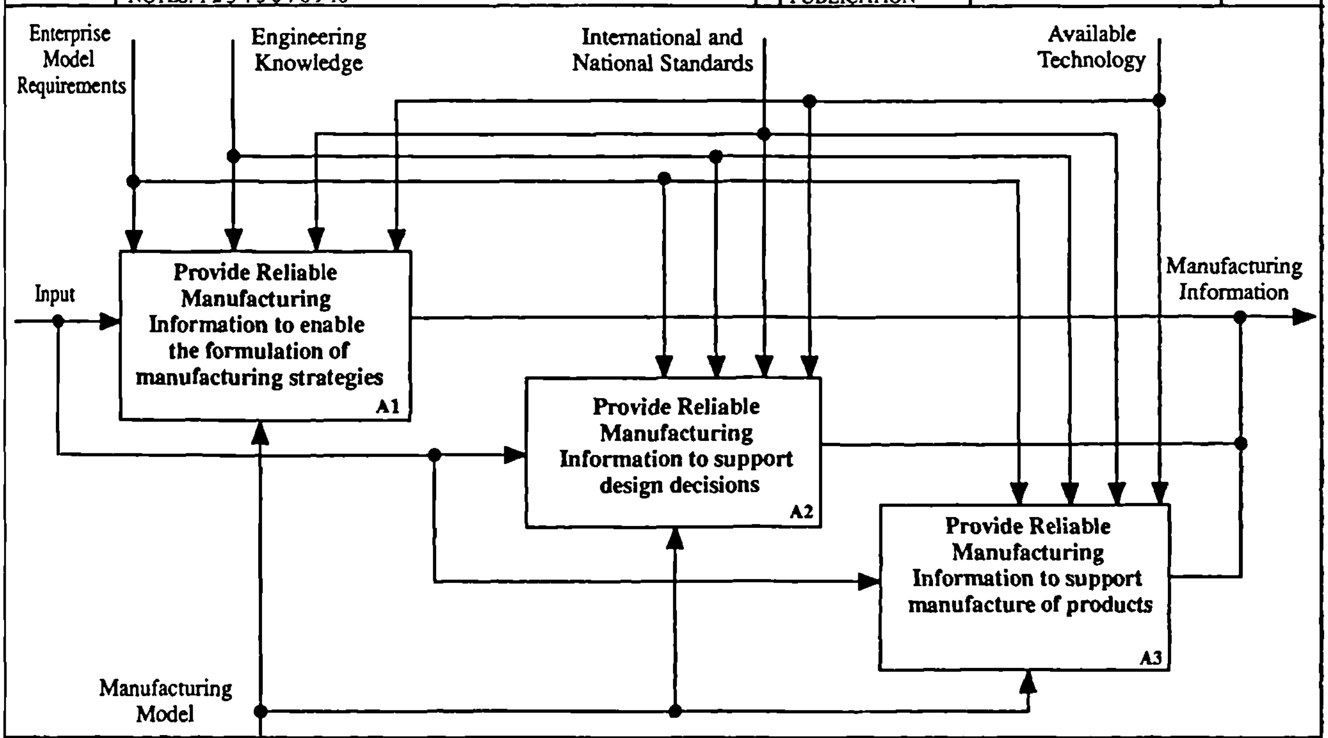
Activity Descriptions

A1 **– Provide Reliable Manufacturing Information to enable the formulation of manufacturing strategies:**
Support Enterprise Activities related to the formulation of new and better manufacturing business strategies by providing information related to the manufacturing capability and manufacturing strategies of the enterprise (AM).

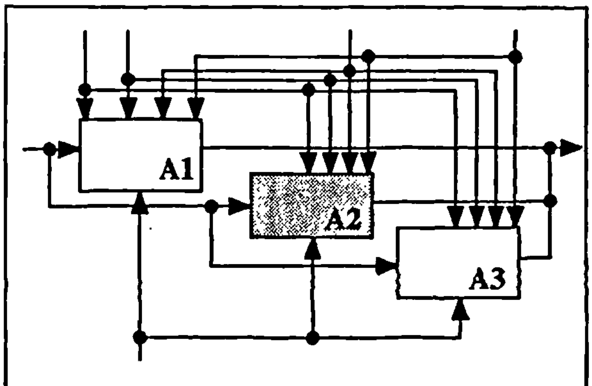
A2 **– Provide Reliable *Manufacturing Information to support design decisions:***
Support Enterprise Activities related to the development of product design by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A3 **– Provide Reliable Manufacturing Information to support manufacture of products:**
Support Enterprise Activities related to the manufacture of products by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	WORKING DRAFT X RECOMMENDED PUBLICATION	READER	DATE	CONTEXT



NODE: A0	TITLE: Provide Reliable Manufacturing Information	NUMBER:

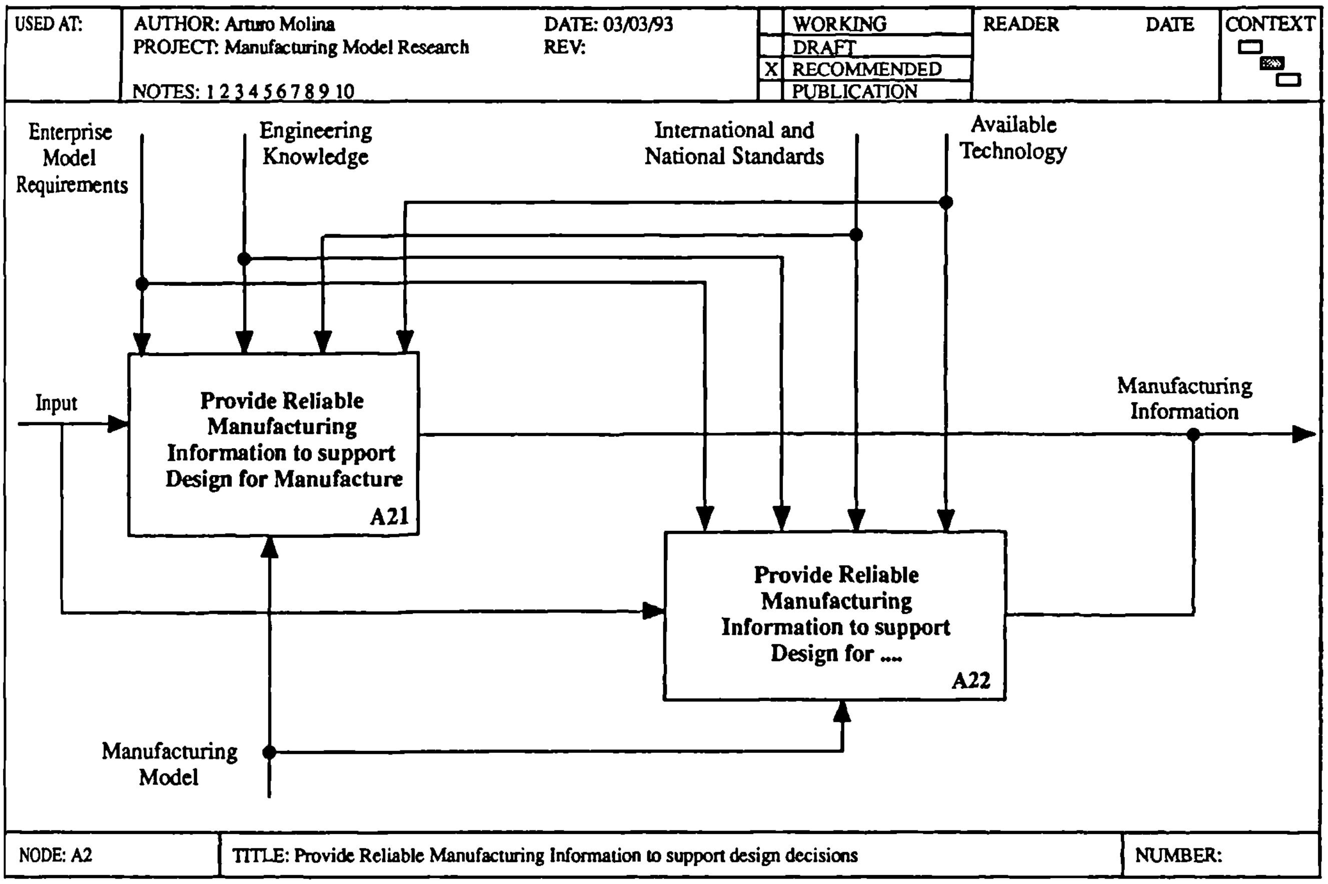


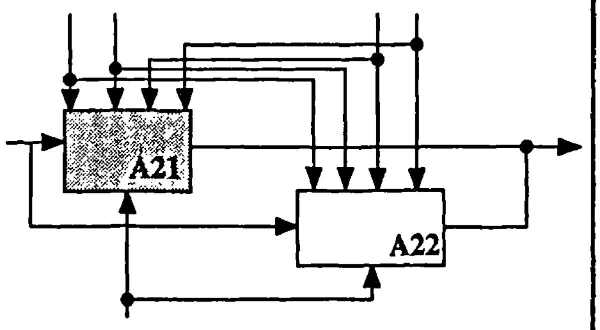
A0 Provide Reliable Manufacturing Information

Activity Descriptions

A21 – **Provide Reliable Manufacturing Information to support Design for Manufacture Activities:**
Support Design for Manufacture Activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A22 – **Provide Reliable Manufacturing Information to support Design for .... Activities:**
Support Design for .... Activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)





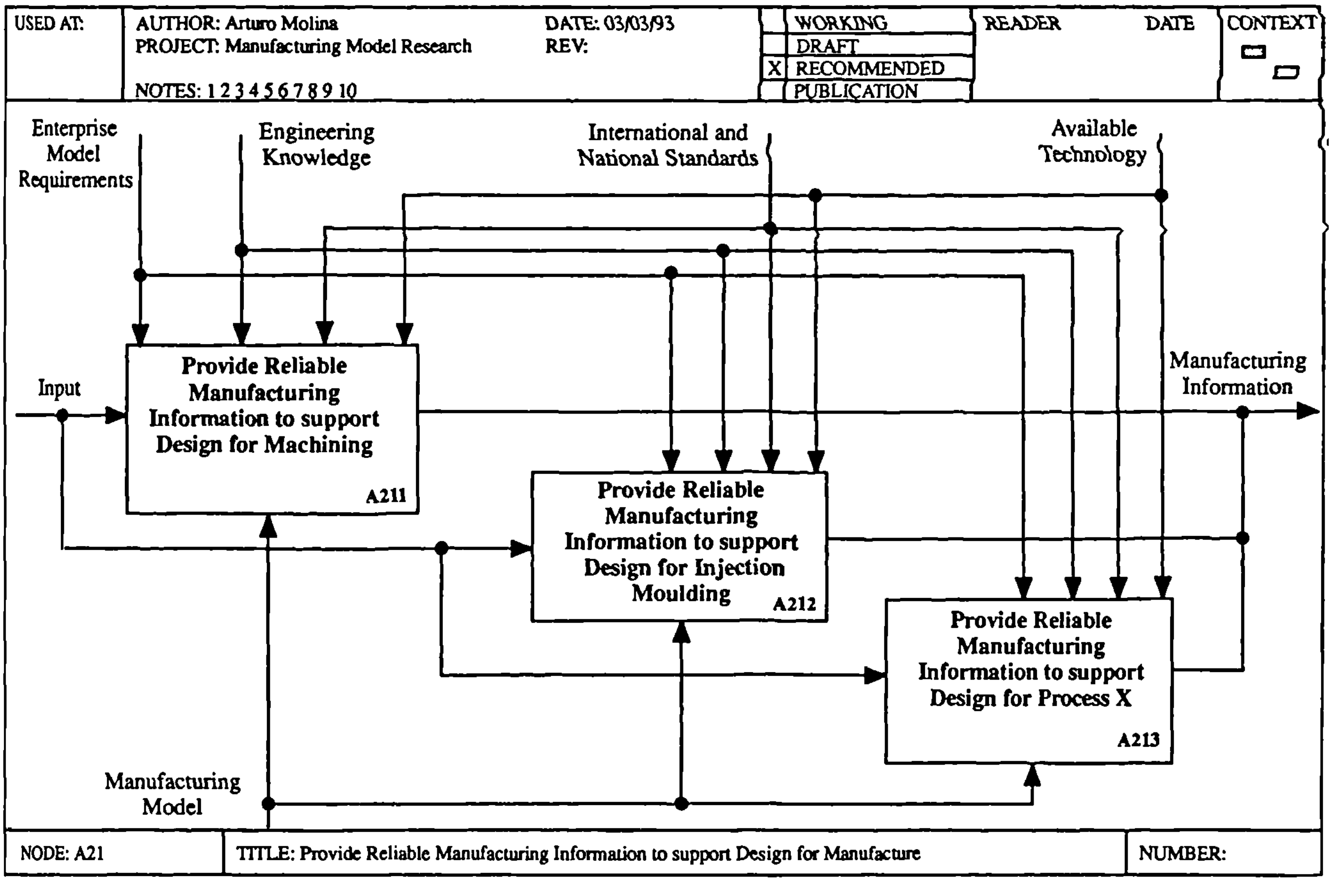
A2 Provide Reliable Manufacturing Information to support design decisions

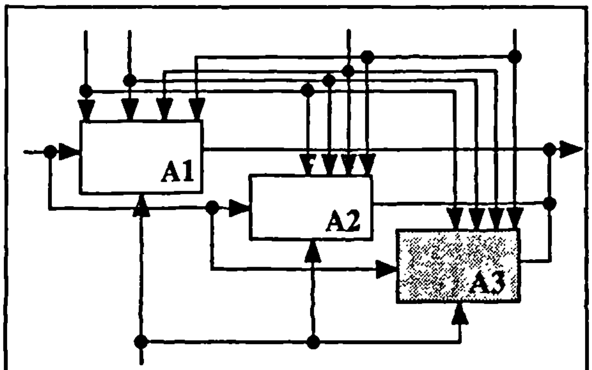
Activity Descriptions

A211 – **Provide Reliable Manufacturing Information to support Design for Machining Activities:**
Support Design for Machining Activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A212 – **Provide Reliable Manufacturing Information to support Design for Injection Moulding Activities:**
Support Design for Injection Moulding Activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A213 – **Provide Reliable Manufacturing Information to support Design for Process X Activities:**
Support Design for Process X Activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)





A0 Provide Reliable Manufacturing Information

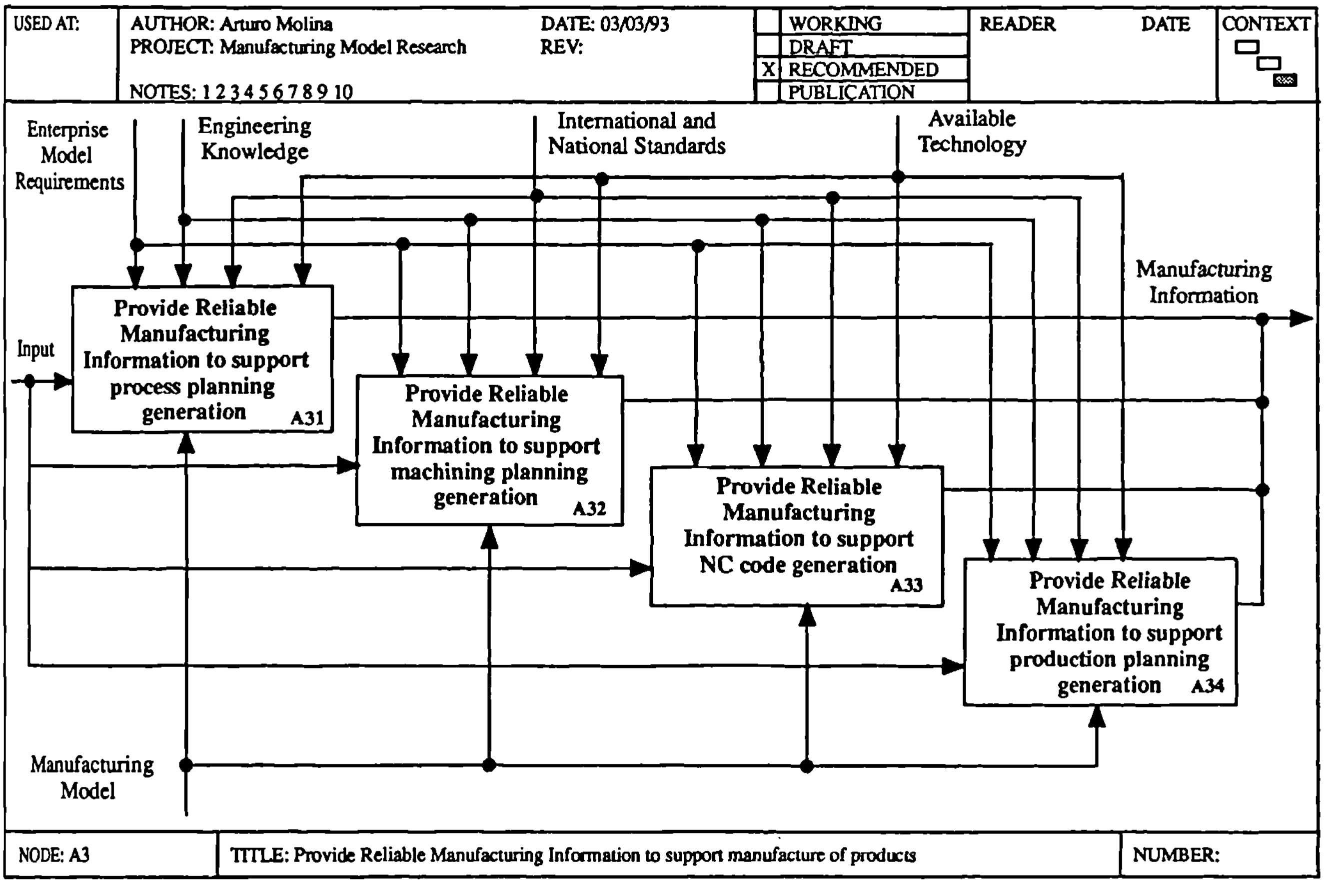
Activity Descriptions

A31 – **Provide Reliable Manufacturing Information to support process planning generation activities:**
Support process planning generation activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A32 – **Provide Reliable Manufacturing Information to support machining planning *generation activities:***
Support machining planning generation activities (NC–Code generation) by providing information related to the manufacturing
capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A33 – **Provide Reliable Manufacturing Information to support pre–production proving:**
Support pre–production proving activities (Inspection Code) by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)

A34 – **Provide Reliable Manufacturing Information to support scheduling generation:**
Support scheduling generation activities by providing information related to the manufacturing capability of the enterprise expressed in terms of manufacturing resources, processes and strategies (AM)



Appendix E

Partial CIMOSA Requirements Definition Model

This appendix presents the partial CIMOSA Requirements Definition Model of the Domain Concurrent Product Development. All the views (function, information, resources, and organisation) are partial developed.

FUNCTION VIEW

DOMAIN

Part 1:	DOMAIN DESCRIPTION
TYPE:	tbd
IDENTIFIER:	DM–01
NAME:	Concurrent Product Development
DESIGN AUTHORITY:	Simultaneous Engineering Team
DOMAIN DESCRIPTION:	Concurrent Product Development is a Domain assigned to design, manufacture and support a product throughout its life–cycle using simultaneous engineering principles and supported by a Computer Aided Simultaneous Engineering System (CAE System).
CIMOSA COMPLIANT:	yes
Part 2:	DOMAIN COMPONENTS
DOMAIN OBJECTIVES:	DO–01/reduce product development time DO–02/improve quality of product DO–03/reduce cost of product
DOMAIN CONSTRAINTS:	DC–01/Business Strategy and Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DOMAIN PROCESSES:	DP–01/NEW PRODUCT SPECIFICATION DP–02/CONCURRENT PRODUCT DESIGN DP–03/MANUFACTURE OF PRODUCTS DP–04/POST–MANUFACTURE ACTIVITIES
DOMAIN BOUNDARY:	RL–01/External Communication
OBJECT VIEWS:	OV–01: New Product order OV–02: Product order OV–03: Post–manufacture order OV–04: Product Specification OV–05: Product Design OV–06: Product
EVENTS:	EV–01: Customer Request EV–02: Specification Release EV–03: Product Design Release EV–04: Product Release EV–05: Event Release

DOMAIN OBJECTIVE/CONSTRAINT

TYPE:	tbd
IDENTIFIER:	DO–01
NAME:	Objective of reduce product development time
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	Domain Processes must be analysed by taking into account the objective of reducing the product development time. Is its expected that the product development time should be reduced between X% and X% of actual time.
SUBJECT:	
TARGET:	
VALUES:	
VALIDITY:	
INHERITED FROM:	

DOMAIN OBJECTIVE/CONSTRAINT

TYPE:	tbd
IDENTIFIER:	DO–02
NAME:	Objective of improving quality of the product
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	Domain Processes must be analysed by taking into account the objective of improving the quality of the product.
SUBJECT:	
TARGET:	
VALUES:	
VALIDITY:	
INHERITED FROM:	

DOMAIN OBJECTIVE/CONSTRAINT

TYPE:	tbd
IDENTIFIER:	DO–03
NAME:	Objective of reduce product cost
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	Domain Processes must be analysed by taking into account the objective of reducing the cost of a product. It is expected that the cost will decrease between X% and X% of the actual cost.
SUBJECT:	
TARGET:	
VALUES:	
VALIDITY:	
INHERITED FROM:	

DOMAIN OBJECTIVE/CONSTRAINT

TYPE: tbd

IDENTIFIER: DC–01

NAME: Business Strategy and Policies

DESIGN AUTHORITY: Manager Director/Simultaneous Engineering Team

DESCRIPTION: Domain Processes must be analysed to always conform with the Business Strategy (e.g. continuous improvement philosophy) and company's policies on outsourcing.

SUBJECT:

TARGET:

VALUES:

VALIDITY:

INHERITED FROM:

DOMAIN OBJECTIVE/CONSTRAINT

TYPE: tbd

IDENTIFIER: DC–02

NAME: Available Technology

DESIGN AUTHORITY: Simultaneous Engineering Team

DESCRIPTION: Domain Processes must be analysed by taking into consideration which technology is available and which technology is the most effective and efficient. New concepts can be used if they are readily available in software applications. Especially attention should be paid in defining in detail the capabilities for a CAE system to support simultaneous engineering.

SUBJECT:

TARGET:

VALUES:

VALIDITY:

INHERITED FROM:

DOMAIN OBJECTIVE/CONSTRAINT

TYPE: tbd

IDENTIFIER: DC–03

NAME: National and International Standards

DESIGN AUTHORITY: Simultaneous Engineering Team

DESCRIPTION: Domain Processes must be analysed by taking into consideration the use of national and international standards whenever possible. For example the use of STEP/EXPRESS for data exchange.

SUBJECT:

TARGET:

VALUES:

VALIDITY:

INHERITED FROM:

DOMAIN OBJECTIVE/CONSTRAINT

TYPE: tbd

IDENTIFIER: DC–04

NAME: Engineering Knowledge

DESIGN AUTHORITY: Simultaneous Engineering Team

DESCRIPTION: Domain Processes must be analysed by involving the most experienced personal available in the different areas involved in the Concurrent Product Development. Personal should include operators, design and manufacturing engineers, managers, etc.

SUBJECT:

TARGET:

VALUES:

VALIDITY:

INHERITED FROM:

ENTERPRISE EVENT:	CUSTOMER REQUEST
TYPE:	tbd
IDENTIFIER:	EV–01
NAME:	Customer Request
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	The Domain Concurrent Product Development is trigger by this event. This event is a request for the development of a new product, production of a catalogued product, or the performance of a post–manufacture support activity (e.g. maintenance or disposal).
GENERATED BY:	External Domain
TRIGGERS:	DP–01/ New Product Specification
RELATED OBJECT:	OV–01: New Product order OV–02: Product order OV–03: Post–manufacture order

ENTERPRISE EVENT:	SPECIFICATION RELEASE
TYPE:	tbd
IDENTIFIER:	EV–02
NAME:	Specification Release
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	The Domain Process Concurrent Product Design is triggered by this event. The release of the Product Specification enables to perform the Concurrent Product Design. This event can trigger the External Domain as well, for example to request the verification of a Product Specification by the customer.
GENERATED BY:	New Product Specification
TRIGGERS:	DP–02/ Concurrent Product Design
RELATED OBJECT:	OV–04: Product Specification

ENTERPRISE EVENT:	PRODUCT DESIGN RELEASE
TYPE:	tbd
IDENTIFIER:	EV–03
NAME:	Product Design Release
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	The Domain Process Manufacture of Products is triggered by this event. The release of Product Designs with the necessary data necessary enables to manufacture products. This event can trigger the External Domain, for example, to request the verification of a Product Design by the customer or to subcontract manufacturing facilities.
GENERATED BY:	Concurrent Product Design
TRIGGERS:	DP–03/ Manufacture of Products
RELATED OBJECT:	OV–05: Product Design

ENTERPRISE EVENT:	PRODUCT RELEASE
TYPE:	tbd
IDENTIFIER:	EV–04
NAME:	Product Release
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	The Domain Process Post–Manufacture Activities is triggered by this event. The release of the Products allows post–manufacture activities to take place, such as maintenance or disposal. This event can trigger the External Domain, for example to inform that a product is available.
GENERATED BY:	Manufacture of Products
TRIGGERS:	DP–02/ Post–Manufacture Activities
RELATED OBJECT:	OV–06: Product

ENTERPRISE EVENT:	EVENT RELEASE
TYPE:	tbd
IDENTIFIER:	EV–05
NAME:	Event Release
DESIGN AUTHORITY:	Simultaneous Engineering Team
DESCRIPTION:	Any event that is released after the completion of a Post–Manufacture Activity and it is outside the Concurrent Product Development Domain.
GENERATED BY:	Post–Manufacture Activities
TRIGGERS:	External Domain
RELATED OBJECT:	tbd

DOMAIN RELATIONSHIP:	EXTERNAL COMMUNICATION
TYPE:	tbd
IDENTIFIER:	RL–01
NAME:	CONCURRENT PRODUCT DEVELOPMENT
DESIGN AUTHORITY:	Simultaneous Engineering Team
DOMAIN DESCRIPTION:	Through this relationship, CONCURRENT PRODUCT REALIZATION Domain receives the order to design, manufacture products or realize some post–manufacture activity related to the Product's Life Cycle.
DOMAIN 1 NAME:	External Domain
INVOLVED OBJECT VIEWS:	OV–01: New Product order OV–02: Product order OV–03: Post–manufacture order OV–04: Product Specification OV–05: Product Design OV–06: Product
INVOLVED EVENTS:	EV–01: Customer Request EV–02: Specification Release EV–03: Product Design Release *EV–04: Product Release* EV–05: Event Release

DOMAIN PROCESS: DP–02, CONCURRENT PRODUCT DESIGN

TYPE:	tbd
IDENTIFIER:	DP–02
NAME:	CONCURRENT PRODUCT DESIGN
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	tbd
DESCRIPTION:	Concurrent Product Design is a Domain Process assigned to design a product while considering all the aspects of its life–cycle using simultaneous engineering principles and supported by a Computer Aided Simultaneous Engineering System (CAE System).
EVENTS:	EV–02: Specification Release EV–03: Product Design Release
FUNCTION INPUTS:	Product_Specification
FUNCTION OUTPUTS:	Product_Design
CONTROL INPUTS:	tbd
CONTROL OUTPUTS:	tbd
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	tbd
ENDING STATUS:	done (design completion)
COMPRISES:	BP–2.1:Management of Design BP–2.2:Design for Life Cycle BP–2.3:Analysis and Test of Design

PROCEDURAL RULES:	WHEN (Event_02)	DO BP–2.1 & BP–2.2
	WHEN (ES(BP–2.2) = done)	DO BP–2.3
	WHEN (ES(BP–2.1) = ok & ES(BP–2.2) = done)	Trigger (Event_03)

BUSINESS PROCESS: BP–2.2, DESIGN FOR LIFE CYCLE

TYPE:	tbd
IDENTIFIER:	BP–2.2
NAME:	DESIGN FOR LIFE CYCLE
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process designs products while considering all the aspects of their life–cycle using simultaneous engineering principles. Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	tbd
FUNCTION OUTPUTS:	Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	tbd
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	tbd
ENDING STATUS:	done (design completion)
COMPRISES:	EA–2.2.1:Clarify Task BP–2.2.2:Create Conceptual Designs BP–2.2.3:Create Embodiment Designs BP–2.2.4:Create Detail Designs
PROCEDURAL RULES:	tbd

BUSINESS PROCESS: BP–2.2.2, CREATE CONCEPTUAL DESIGNS

TYPE:	tbd
IDENTIFIER:	BP–2.2.2
NAME:	CREATE CONCEPTUAL DESIGNS
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process which designs viable concepts for products based on the product specification. Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	
FUNCTION OUTPUTS:	Selected_Schemes
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	tbd
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	tbd
ENDING STATUS:	done (design completion)
COMPRISES:	EA–2.2.2.1:Establish Function Structures EA–2.2.2.2:Search for Solution Principles EA–2.2.2.3:Evaluate Concepts
PROCEDURAL RULES:	tbd

BUSINESS PROCESS: BP–2.2.3, CREATE EMBODIMENT DESIGNS

TYPE:	tbd
IDENTIFIER:	BP–2.2.3
NAME:	CREATE EMBODIMENT DESIGNS
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process designs products by developing their basic characteristics while considering all the aspects of their life cycle using simultaneous engineering principles. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Selected_Schemes
FUNCTION OUTPUTS:	Preliminary_Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	
ENDING STATUS:	done (design completion)
COMPRISES:	EA–2.2.3.1:Do Design for Function BP–2.2.3.2:Evaluate Embodiment Designs
PROCEDURAL RULES:	

BUSINESS PROCESS: BP–2.2.4, CREATE DETAIL DESIGNS

TYPE:	tbd
IDENTIFIER:	BP–2.2.4
NAME:	CREATE EMBODIMENT DESIGNS
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process designs products by developing their detail description while considering all the aspects of their life cycle using simultaneous engineering principles. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Preliminary_Product_Design
FUNCTION OUTPUTS:	Product_Design
CONTROL INPUTS:	Product_*Specification*
CONTROL OUTPUTS:	none
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	BP–2.2.4.1:Create Detail Design Data BP–2.2.4.2:Evaluate Detail Designs
PROCEDURAL RULES:	

BUSINESS PROCESS: BP–2.2.3.2, EVALUATE EMBODIMENT DESIGNS

TYPE:	tbd
IDENTIFIER:	BP–2.2.3.2
NAME:	EVALUATE EMBODIMENT DESIGNS
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process evaluates embodiment designs by considering all the aspects of the product's life cycle using simultaneous engineering principles. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Preliminary_Product_Design
FUNCTION OUTPUTS:	Preliminary_Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	none
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	EA–2.2.3.2.1:Do Design for Manufacture BP–2.2.3.2.2:Do Design for X
PROCEDURAL RULES:	

BUSINESS PROCESS: BP–2.2.4.1, CREATE DETAIL DESIGN DATA

TYPE:	tbd
IDENTIFIER:	BP–2.2.4.1
NAME:	CREATE DETAIL DESIGN DATA
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process creates the detail design data. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Preliminary_Product_Design
FUNCTION OUTPUTS:	Preliminary_Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	none
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	tbd
PROCEDURAL RULES:	

BUSINESS PROCESS: BP–2.2.4.2, EVALUATE DETAIL DESIGN

TYPE:	tbd
IDENTIFIER:	BP–2.2.4.2
NAME:	EVALUATE DETAIL DESIGNS
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process evaluates detail designs by considering all the aspects of the product's life cycle using simultaneous engineering principles. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Preliminary_Product_Design
FUNCTION OUTPUTS:	Preliminary_Product_Design *Product_Design*
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	tbd
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	EA–2.2.4.2.1: Do Design for Manufacture BP–2.2.4.2.2: Do Design for X
PROCEDURAL RULES:	

BUSINESS PROCESS: BP–2.2.3.2.2, DO DESIGN FOR X

TYPE:	tbd
IDENTIFIER:	BP–2.2.3.2.2
NAME:	DO DESIGN FOR X
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Business Process evaluates the product design by considering how different aspects of the product's life cycle (e.g. maintenance, recycling, disposal) influences product design. This Business Process is supported by a Computer Aided Simultaneous Engineering System (CAE System).
FUNCTION INPUTS:	Preliminary_Product_Design
FUNCTION OUTPUTS:	Preliminary_Product_Design
CONTROL INPUTS:	Product_Specification
CONTROL OUTPUTS:	none
RESOURCE INPUTS:	CAE System
RESOURCE OUTPUTS:	none
ENDING STATUS:	done (design completion)
COMPRISES:	tbd
PROCEDURAL RULES:	

ENTERPRISE ACTIVITY: EA–2.2.3.1, DO DESIGN FOR FUNCTION

TYPE:	tbd
IDENTIFIER:	EA–2.2.3.1
NAME:	DO DESIGN FOR FUNCTION
DESIGN AUTHORITY:	Simultaneous Engineering Team

A. Functional Description

OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Apply Design for Function Rules to the Selected Scheme to create a preliminary product design by considering how the layout of a product can achieve the required functionality. On the other hand apply the same rules to the Preliminary Product Design in order to check if the required functionality is alright. This Enterprise Activity is supported by a Computer Aided Simultaneous Engineering System (CAE System).

INPUTS

FUNCTION INPUT:	OV–07/ Selected_Schemes OV–08/Preliminary_Product_Design
CONTROL INPUTS:	OV–04/Product_Specification
RESOURCE INPUTS:	CAE System

OUTPUTS:

FUNCTION OUTPUTS:	OV–08/Preliminary_Product_Design
CONTROL OUTPUTS:	EV–06/Design_Change
RESOURCE OUTPUTS:	none
ENDING STATUS:	completion status: done (preliminary design)
REQUIRED CAPABILITIES:	CAE System Capabilities (Design for Function Application)

ENTERPRISE ACTIVITY: EA–2.2.3.2.1, DO DESIGN FOR MANUFACTURE

TYPE:	tbd
IDENTIFIER:	BP–2.2.3.2.1
NAME:	DO DESIGN FOR MANUFACTURE
DESIGN AUTHORITY:	Simultaneous Engineering Team
OBJECTIVES:	DPO–01/reduce product design time DPO–02/improve quality of product DPO–03/reduce cost of product design
CONSTRAINTS:	DC–01/Design Policies DC–02/Available Technology DC–03/National and International Standards DC–04/Engineering Knowledge
DECLARATIVE RULE:	none
DESCRIPTION:	Apply Design for Manufacture Rules to the Preliminary Product Design to evaluate how the way in which a product will be manufactured influences the product design. Design changes might be suggested in order to improve the product design. This Enterprise Activity is supported by a Computer Aided Simultaneous Engineering System (CAE System).
INPUTS	
FUNCTION INPUT:	OV–08/Preliminary_Product_Design
CONTROL INPUTS:	OV–04/Product_Specification OV–09/Manufacturing Capabilities OV–10/DFM Rules
RESOURCE INPUTS:	CAE System
OUTPUTS:	
FUNCTION OUTPUTS:	OV–08/Preliminary_Product_Design
CONTROL OUTPUTS:	EV–06/Design_Change
RESOURCE OUTPUTS:	none
ENDING STATUS:	completion status: done (preliminary design)
REQUIRED CAPABILITIES:	CAE System Capabilities (Design for Manufacture Application)

RESOURCE VIEW

Template: CAE SYSTEM CAPABILITY

TYPE:	Information System
IDENTIFIER:	CAE
NAME:	Computer Aided Simultaneous Engineering System
DESCRIPTION:	Provides computer support to the Simultaneous Engineering team throughout the life cycle of the product
CAPABILITIES:	Support the Enterprise Activity: EA–2.2.3.2.1 Do Design for Manufacture By giving design advise on process choice By giving design advise on manufacturability By providing a set of Design for Manufacturing Rules Provide Reliable Information to Support the Enterprise Activity: EA–2.2.3.2.1 Do Design for Manufacture Information related to the product specification, geometry and manufacturing data. Information related to the manufacturing capability of a process, resource or facility.

INFORMATION VIEW

OBJECT VIEW: OV–08, Preliminary Product Design

TYPE:	Product Information
IDENTIFIER:	OV–08
NAME:	Preliminary Product Design

DESIGN AUTHORITY: Simultaneous Engineering Team

NATURE:	Information
DESCRIPTION:	Description of the Product Design Information as it evolves
PROPERTIES:	Geometry Function Features Form Features Manufacturing Data
CONSTRAINTS:	

OBJECT VIEW: OV–09, Manufacturing Capability Information

TYPE:	Manufacturing Information
IDENTIFIER:	OV–09
NAME:	Manufacturing Capability Information
DESIGN AUTHORITY:	Simultaneous Engineering Team
NATURE:	Information
DESCRIPTION:	Description of the Manufacturing Capability Information of a Facility
PROPERTIES:	Resource Type and Configurations Process Type Facility Type and Configurations Resource Capability: Physical Limitations Process Capability: Physical Limitations Resources: Cost, Times, Availability Process: Cost, Times, Availability
CONSTRAINTS:	Companies' Operational Rules

ORGANISATION VIEW

TYPE:	Team
IDENTIFIER:	SE–Team
NAME:	Simultaneous Engineering Team
DESIGN RESPONSIBILITY:	Identify the requirements for implementing Simultaneous Engineering in the Company using a CAE System
DESIGN AUTHORITY:	Validate the requirements
DESCRIPTION:	The SE–Team is the responsible for developing the model for CAE requirements to support Simultaneous Engineering. The team should consider all the important aspect, especially the information required in Simultaneous Engineering.

Appendix F

IDEF0 Model of the Information Viewpoint of the Manufacturing Model

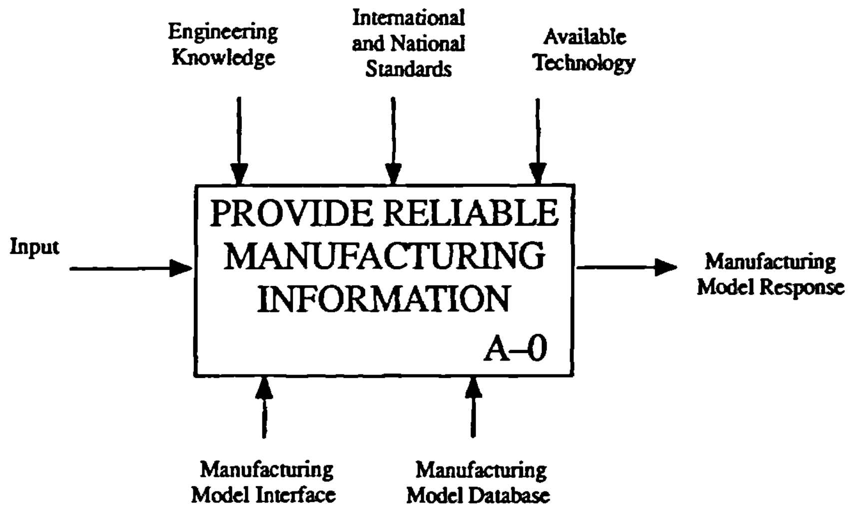
This appendix presents the IDEF0 Model which is part of the Information Viewpoint of the Manufacturing Model. The other component of the Information Viewpoint is the EXPRESS Model. This IDEF0 Model is related to the Enterprise Viewpoint of the Manufacturing Model.

Activity Descriptions

A0 **– Provide Reliable Manufacturing Information:**

Support Enterprise Activities and CAE System elements by providing information related to the manufacturing capabilities of the enterprise that is unambiguous, consistent and up to date (AM)

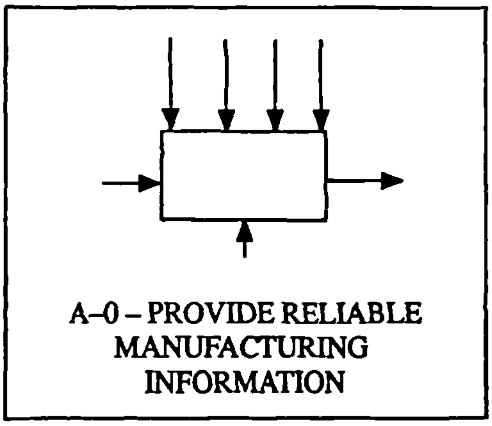
USED AT:	AUTHOR: Arturo Molina PROJECT: Manufacturing Model Research NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 03/03/93 REV:	[ ] WORKING [ ] DRAFT [X] RECOMMENDED [ ] PUBLICATION	READER	DATE	CONTEXT Top



Purpose: to model the functions required in the Manufacturing Model to provide the manufacturing information to the system users.

Viewpoint: a System Programmer which intents to specify the Manufacturing Model functions

NODE: A-0	TITLE: Provide Reliable Manufacturing Information	NUMBER:



Activity Descriptions

A1 **– Execute User Interface:**

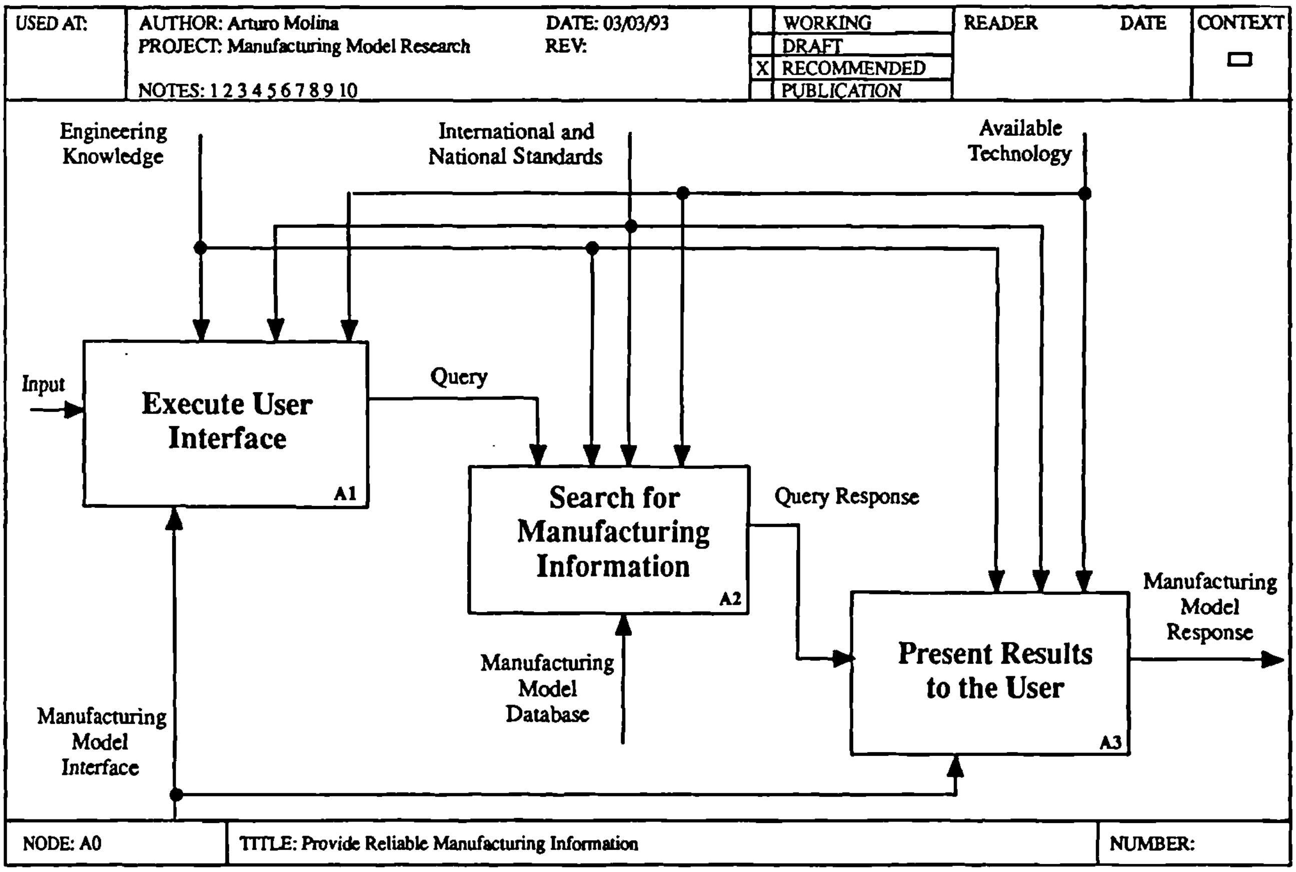
Execute the Manufacturing Model Interface of the Manufacturing Model to allow the user to access the manufacturing information captured in the Manufacturing Model (AM).

A2 **– Search for Manufacturing Information:**

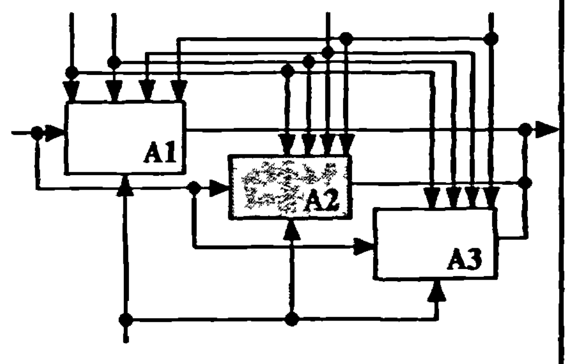
Execute the functions (methods) to access the information of the Manufacturing Model Database requested by the user (AM).

A3 **– Present Results to User:**

Execute the Manufacturing Model Interface of the Manufacturing Model to display the results of the user's requests



Activity Descriptions



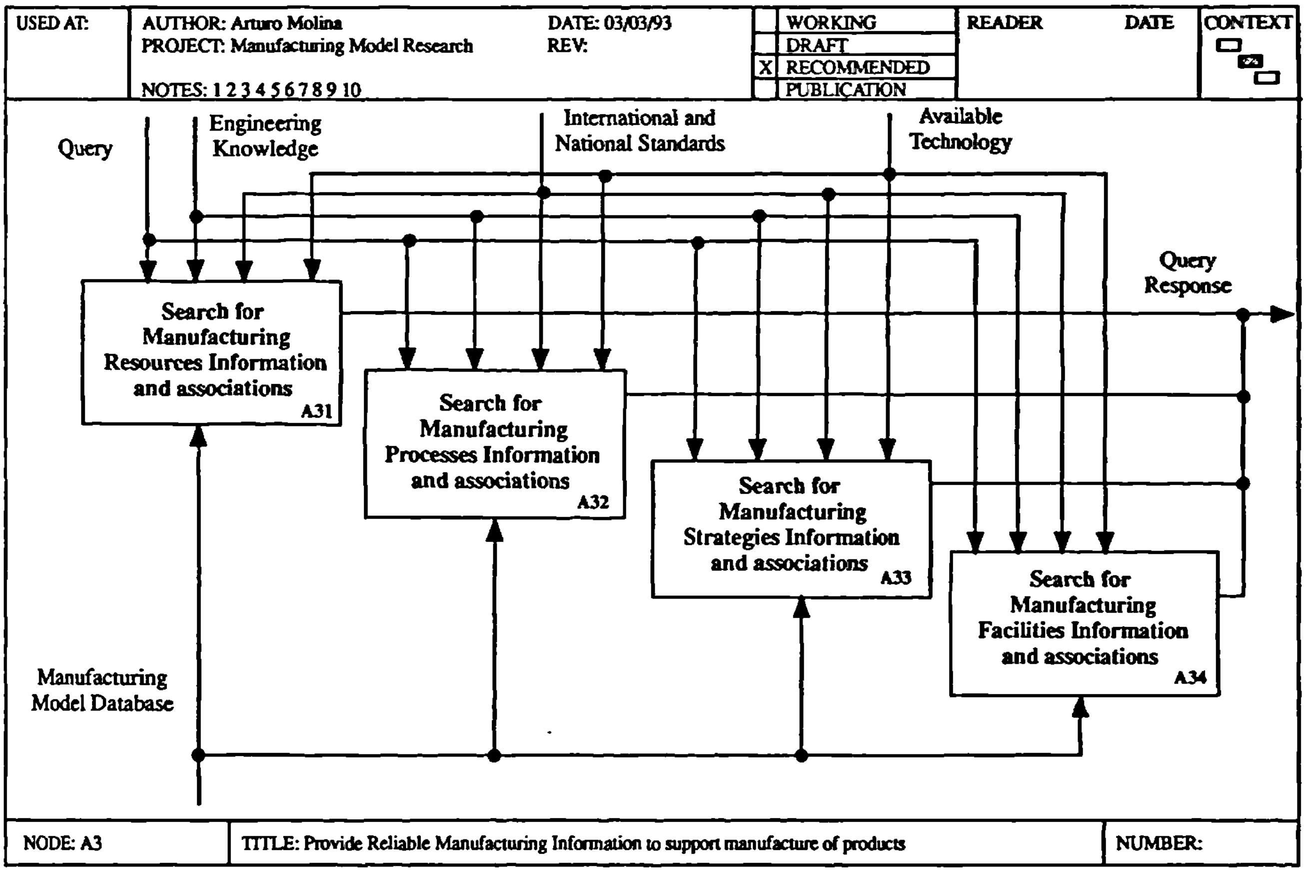
A0 Provide Reliable Manufacturing Information

A21 **– Search for Manufacturing Resources Information and associations :**
Search in the Manufacturing Model Database the information regarding the Manufacturing Resources and associated Manufacturing Processes and Strategies (AM)

A22 **– Search for Manufacturing Processes Information and associations :**
Search in the Manufacturing Model Database the information regarding the Manufacturing Processes and associated Manufacturing Resources and Strategies (AM)

A22 **– Search for Manufacturing Strategies Information and associations :**
Search in the Manufacturing Model Database the information regarding the Manufacturing Strategies and associated Manufacturing Resources and Processes (AM)

A22 **– Search for Manufacturing Facilities Information and associations :**
Search in the Manufacturing Model Database the information regarding the Manufacturing Facilities (Factory, Shop Floors, Cells and Stations) and associated Manufacturing Resources, Processes and Strategies (AM)



Appendix G

Visits to the Mazak European Factory at Worcester

G1. Introduction

This Appendix contains the major tasks carried out by the author to develop the case studies presented in Chapters 13, 14 and 15. This appendix also offers a record of the major points raised in conversations carried out during three visits to the Mazak European Factory.

G2. The Tasks of the Case Study

The following tasks were carried out by author to develop this case study:

1. Identification of the facilities and related resources and processes to model and collect all the information available using public domain brochures of the Yamazaki Company.

2. Selection of the generic resources and processes described in Chapter 12 which better represent the resources and processes described in the brochures.

3. Selection of the aggregation relations *(has)*, semantic relations *(performs, holds, etc.)* and generic classes which have to be instantiated.

4. Instantiation of the administrative, technical and operational attributes of the generic classes in the Manufacturing Model prototype.

5. Creation of an instance at the Station Level to configure the Rotational Parts Stations and their related resources and processes.

6. Creation of an instance at the Cell Level to configure the Rotational Parts Line and its related resources and processes.

7. Carried out three visits to the Yamazaki Factory at Worcester to interview employees to verify information and build up a set of common operational rules.

8. Modification of the instances in the Manufacturing Model prototype according to information revised in the visits.

9. Instantiation of the strategies based on the operational rules collected during the visits.

10. Use of the Manufacturing Model prototype to support the activities of Design of Manufacture and NC Planning.

G3. First Visit

After the modelling work related to the Rotational Parts Stations and associated resources and processes was completed (tasks 1 to 5), the first visit to the Worcester Factory was carried out to verify and complement the information captured in the model.

In an interview with Mr. Tim Hoobs (Supervisor Machining) and Mr. Ashley Cross (Team Leader of Rotational Parts) the model was validated and some operational rules were drawn based on their experience. An example workpiece was offered to allow the author to represent a realistic example in this case study. This workpiece has been used in the demonstration of the NC Planning application.

From this interview the author recognised that quite a lot of the attributes defined to represent Turning Centres and related components were not significant, and therefore in the instances of the database these attributes were recorded as zeros. In addition, the jargon used in the factory was introduced to the model in order to avoid future misunderstandings. For example, the author defined "driven tools", where in the factory these tools are called "life tools". On the other hand, the position of the tools in relation to the workpiece were called "vertical" (perpendicular) and "horizontal" (parallel). In relation to the tooling system, the tool assemblies are simply called "tools", where each tool combines a tool holder, cutter and the "tip" (insert).

This visit was very useful to prove the correctness and completeness of the generic classes defined in the Manufacturing Model (Chapter 12). Finally, the information gathered in this visit confirmed that no major changes were required to the structure of the Manufacturing Model at the Station Level.

G4. Second Visit

A model of the Rotational Parts Line and associated resources and processes (tasks 1 to 5) was build previously to this visit . This visit had the objective to study the operation of the Rotational Part Lines and to try to capture some operational rules which control its function.

The first interview was with Mr. Gary Whitehouse (Shop Floor Manager) who gave the author an introduction to the general concepts of how the production control is carried out and how the Rotational Parts Line operates. In order to gain more understanding the author discussed details of the operation with Mr. Ashley Cross (Team Leader of Rotational Parts).

As the author wanted to know a little bit more about the general concepts behind the production control, we were turned to Mr. Richard Austin (Production Controller), who explained how the production in the factory is carried out.

In general, all these interviews gave the author a very good insight of how the Rotational Parts Line is organised, operated and what is its role within the factory production process. This visit allow the author to build the set of realistic strategic decisions and operational rules for the Rotational Parts Line model presented in Chapter 15, Section 15.5.

G5. Third Visit

This last visit had the objective to analyse the Assembly Hall, understand its organisation and explore the possibilities to model it in the Manufacturing Model. The Assembly Manager Mr. Richard Smith explained how the Assembly Line is operated and how it is related to other manufacturing process within the factory. This visit allowed the author to have a better understanding of the assembly process in other to be able to offer some conclusions on how to model this process in the Manufacturing Model (15, 15.7).

Appendix H

Instances of the Rotational Parts Station

H.1. Introduction

This appendix describes the instances of the generic classes used in Chapter 13 to represent the model of the Rotational Parts Station in the Manufacturing Model Database.

H.2. Turning Centre and Machine Components Instances

The schema instantiated of the TURNING CENTRE class is presented in figure H.1. Values of zeros indicate attributes which are not important. Figure H.1 shows the relation *has* between the TURNING CENTRE and MACHINE COMPONENTS. This relation is created while populating the model of the Slant Turn 40N ATC Mill Center, and therefore the instances MACHINE COMPONENT created are: SPINDLE, TURRET (TOOL CARRIER), TOOL MAGAZINE (TOOL CARRIER) and CONTROLLER. The schema of the SPINDLE is illustrated in figure H.3 and its instance in figure H.4, where the important attributes are the axis of motion, spindle clamp diameter, spindle bore diameter and maximum speed. It should be noted that the SPINDLE is a MACHINE TOOL COMPONENT and a MOVEABLE AXIS COMPONENT, this allows the definition of motions. The TURRET class is presented in figure H.5. This class defined the *holds* relation which enables to represent the capability of the TURRET to hold production tools. In the instance of this class (figure H.6) the turret of the Slant Turn 40N ATC Mill Center can hold the following tool holders FACE MILL ARBOR, MORSE TAPER HOLDER, CHUCK HOLDER, BORING BAR HOLDER, CUTTING TOOL HOLDER and U DRILL HOLDER. This definitions are according to the figure 13.4 and 13.5 presented in Chapter 13. The other two components the TOOL MAGAZINE and CONTROLLER are illustrated in figures H.7, H.8, H.9 and H.10.

H.3. Tool System Instances

The instances of the tooling system were simplified due to the fact that in the Worcester factory it is the important to represent which tools you can use, and not how they are composed. Therefore even when the model developed uses the generic class GENERIC TOOL HOLDER (see Chapter 13, Subsection 13.2.2), this class was not instantiated; instead the more generic class of TOOL HOLDER was used, together with the CUTTER class. The two type of tool holders required to hold static and life tools (driven) were represented using the TOOL HOLDER class. For life tools different instances of the following holders where defined FACE MILL ARBOR, MORSE TAPER HOLDER, CHUCK HOLDER (Figure H.11, H12 and H13). These instances are related to the TURRET component of the Slant Turn 40N ATC Mill Center. All these tool holders are related to the cutting tools using the relation *holds*. Figure H.11 shows that the FACE MILL ARBOR *holds* a FACE MILL, figure H.12 that MORSE TAPER HOLDER *holds* a BORING BAR and, figure H.13 that CHUCK HOLDER *holds* a CENTER DRILL. In a similar way the holders for static tools BORING BAR HOLDER (figure H.14) *holds* BORING BAR and BORING BAR WITH INSERTS, CUTTING TOOL HOLDER (figure H.15) *holds* CUTTING TOOL 25x25, and U DRILL HOLDER (figure H.16) *holds* U DRIL. All these instances represent the model of the tool capabilities of the Slant Turn 40N ATC Mill Center described in figures 13.4 and 13.5.

Each individual tool has an instance which is composed of three or two elements depending on its type. Indexable insert tools are composed by the basic definition, type definition and insert holder definition. For example the FACE MILL is an indexable insert cutter therefore it is composed by FACE MILL instance (basic definition), FACE MILL type (type definition) and FACE MILL insert holder definition, see figure H.17. The basic definition has attributes related to the type of insert, tool geometry, and too movements. It also indicates which tool holder is holding this tool. The type definition indicates which type is and enable to make an association to the insert holder definition. This last definition describe the details of the tool based on a ISO/BSI standard, in this case the BS4193 part 17. This structure was implemented in order to reuse in a better way the class definitions. In the case of solid tools, these are composed only of two definition, basic and type definition. The example of the DRILL in figure H.18 shows the basic definition indicating that is a solid cutter and the DRILL type definition with detailed information according to the standard BS 328. The other tools of the model are shown in figures H.19, H.20 and H.21.

H.4. Machining Processes Instances

The generation of the instances for machining processes employs the definition of the class CHIP FORMING PROCESS, and requires to create the relations "requires_workpiece" and "requires_tool". The respective workpiece instance and set of tools instances are associated to the process using this relation. Because it is not important to identify the workpiece and tools (the definition used by the processes), identifiers are generated automatically by the database. In the figure H.22 for example the identifiers #2–2–3–29 and #2–2–3–32 represent the workpiece and tool required to describe the process capabilities. The instance illustrated in figure H.22 represent the Out Side Diameter Turning Process which requires a workpiece to have movement on Rz (free rotational movement) and a tool (or set of tools) with movements on X and Z axis. The tool has to be for machining external surfaces, with vertical position in relation to the workpiece and static. Figure H.23 shows the characteristics of the workpiece and type of tool required to perform the OD Turning Process. The workpiece has to rotate in the Rz axis, and the tool is or the type of the CUTTING TOOL 25x25.

H.5. Turning Centre Process Capabilities Instances

In order to represent the machining capabilities of the Slant Turn 40N ATC Mill Center, the relation "performs" has to be created between this turning center and the different machining processes created. This relation will allow the users of the Manufacturing Model to search for the processes capabilities of a station by inquiring for the processes the Slant Turn 40N ATC Mill Center is able to perform (Figure H.24)

H.6. Rotational Parts Station Configuration Instance

The Station Level of the Manufacturing Model is employed to create an instance of the Rotational Parts Station. In addition to this instance and the one of the Slant Turn 40N ATC Mill Center, two other elements have been created to build a more complete definition of the station, i.e. the instances of the Mazak Loading Robot and Pallet Buffer (Figure H.25). All these instances represent the Manufacturing Model populated, and therefore this Manufacturing Model is ready to be used by data–driven applications.

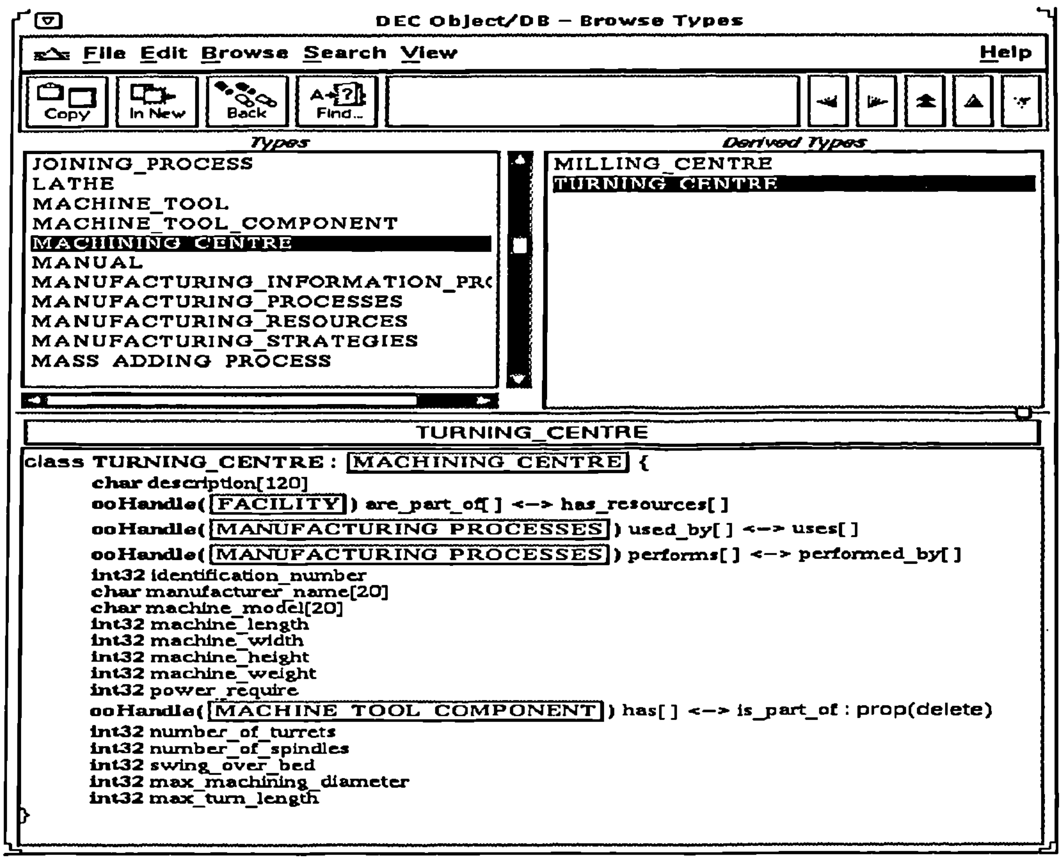


Figure H.1 Turning Centre Schema

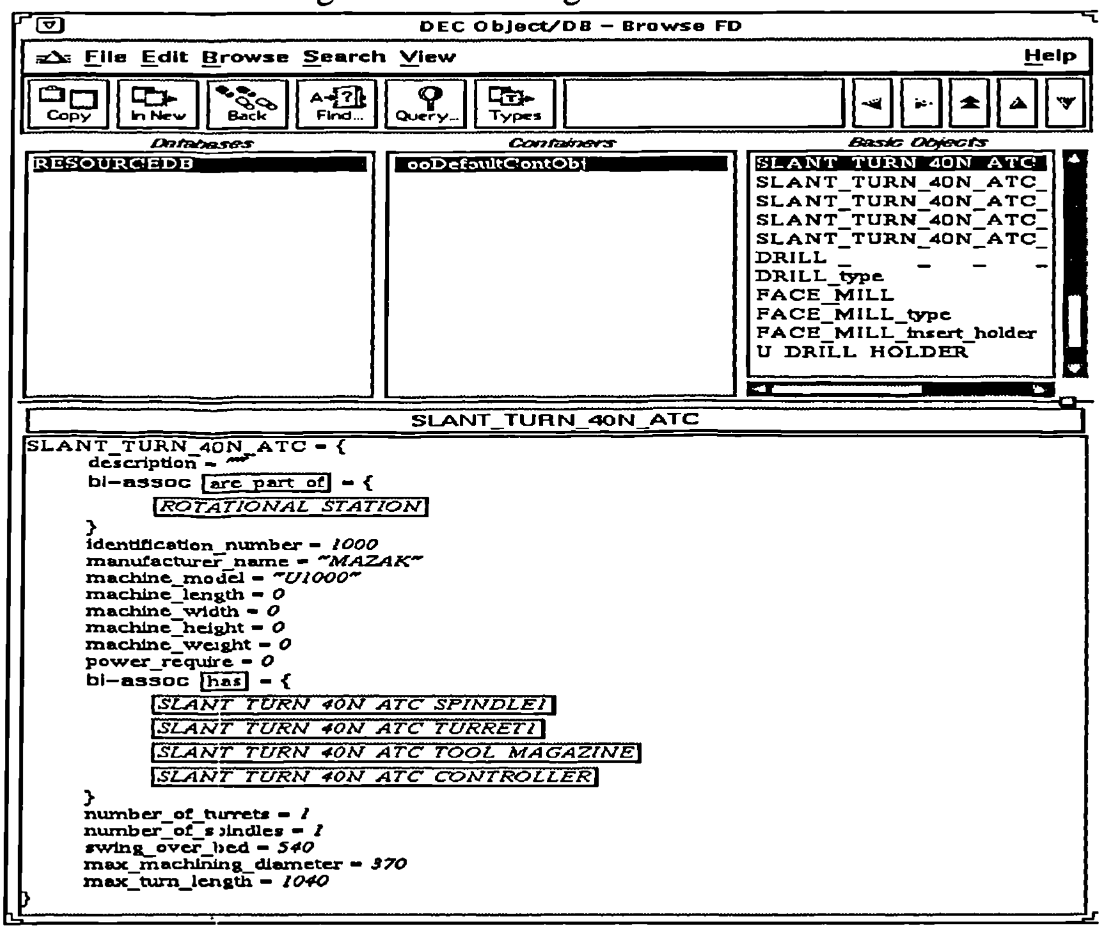


Figure H.2 Slant Turn 40N ATC Mill Centre Instance

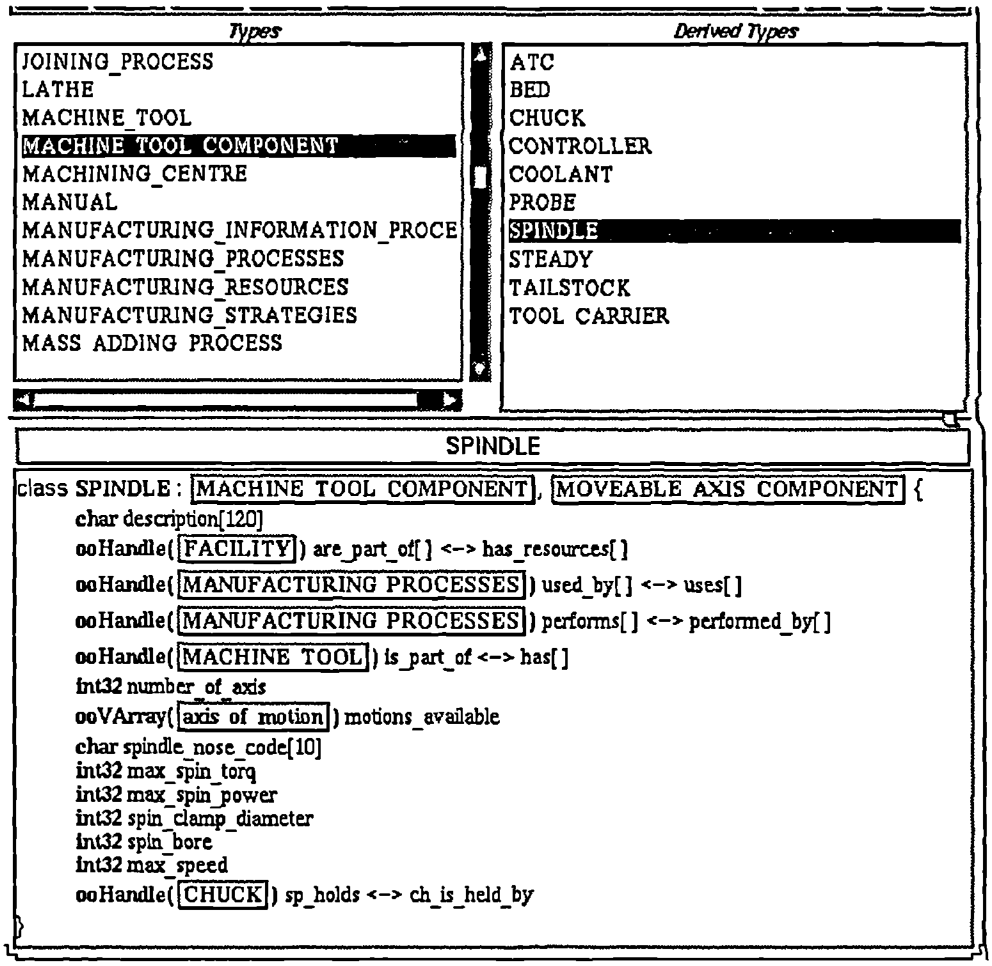


Figure H.3 Spindle Schema

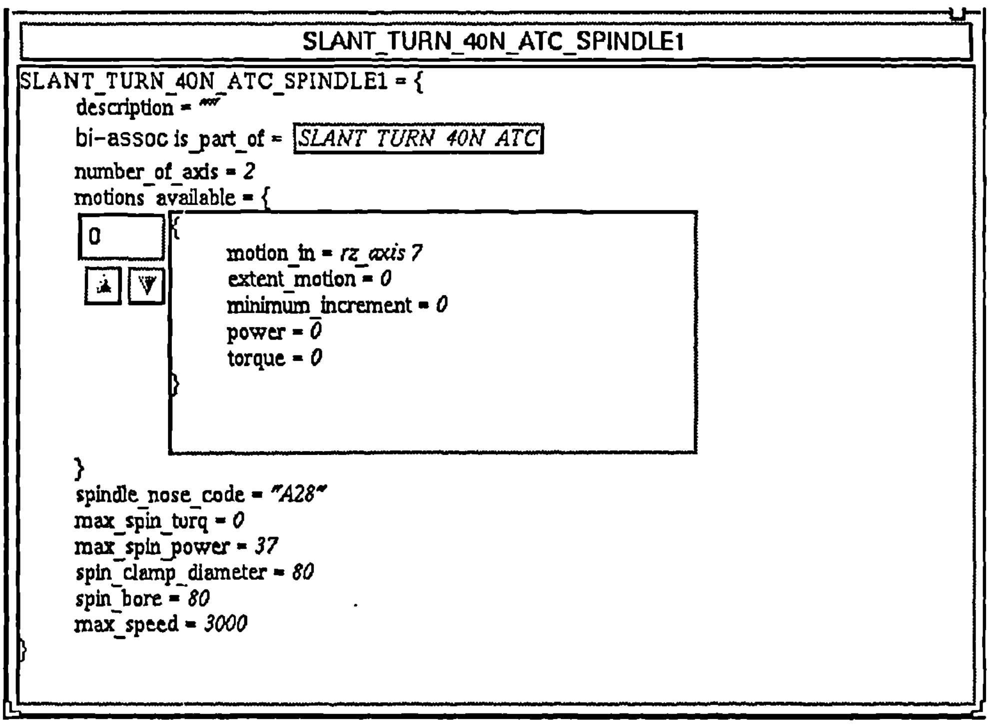


Figure H.4 Slant Turn 40N ATC Mill Centre Spindle Instance

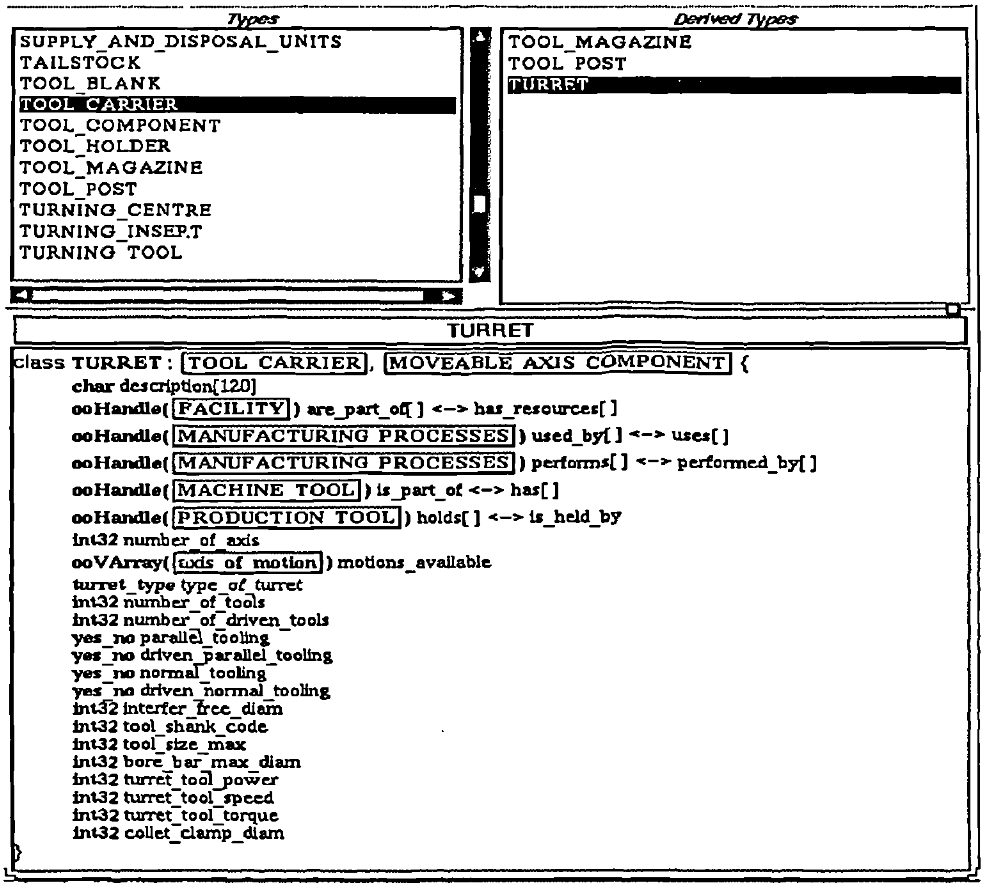


Figure H.5 Turret Schema

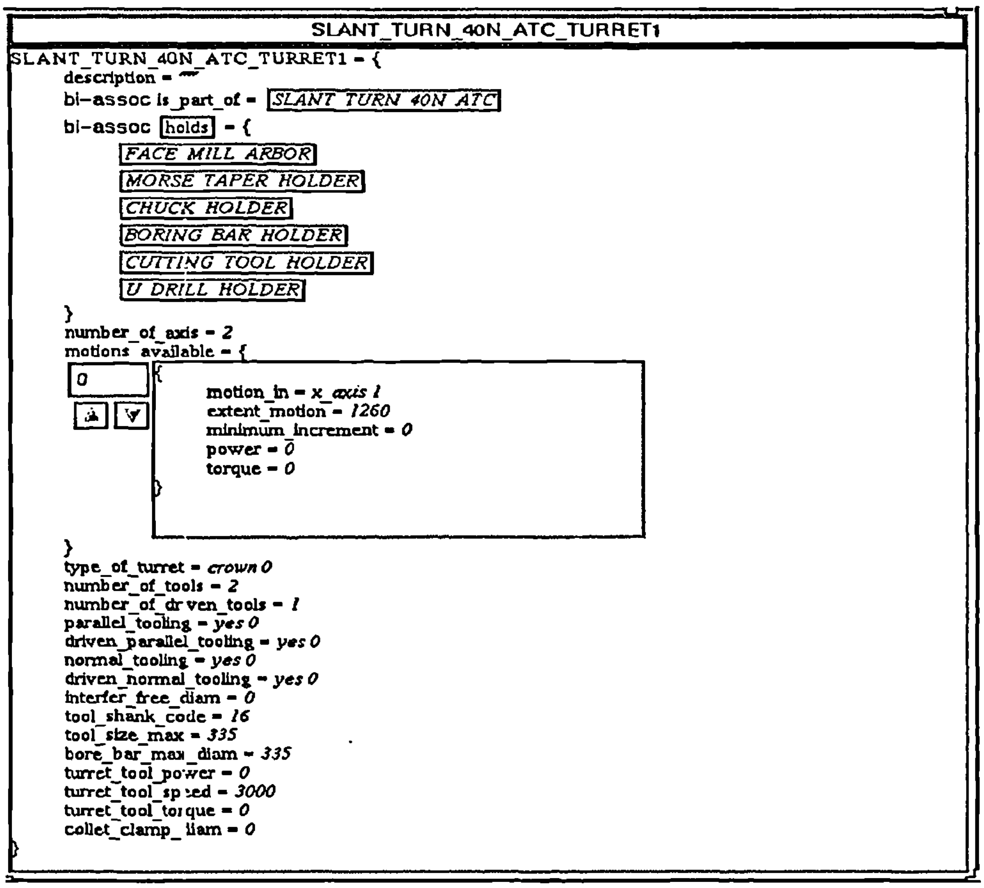


Figure H.6 Slant Turn 40N ATC Mill Centre Turret Instance

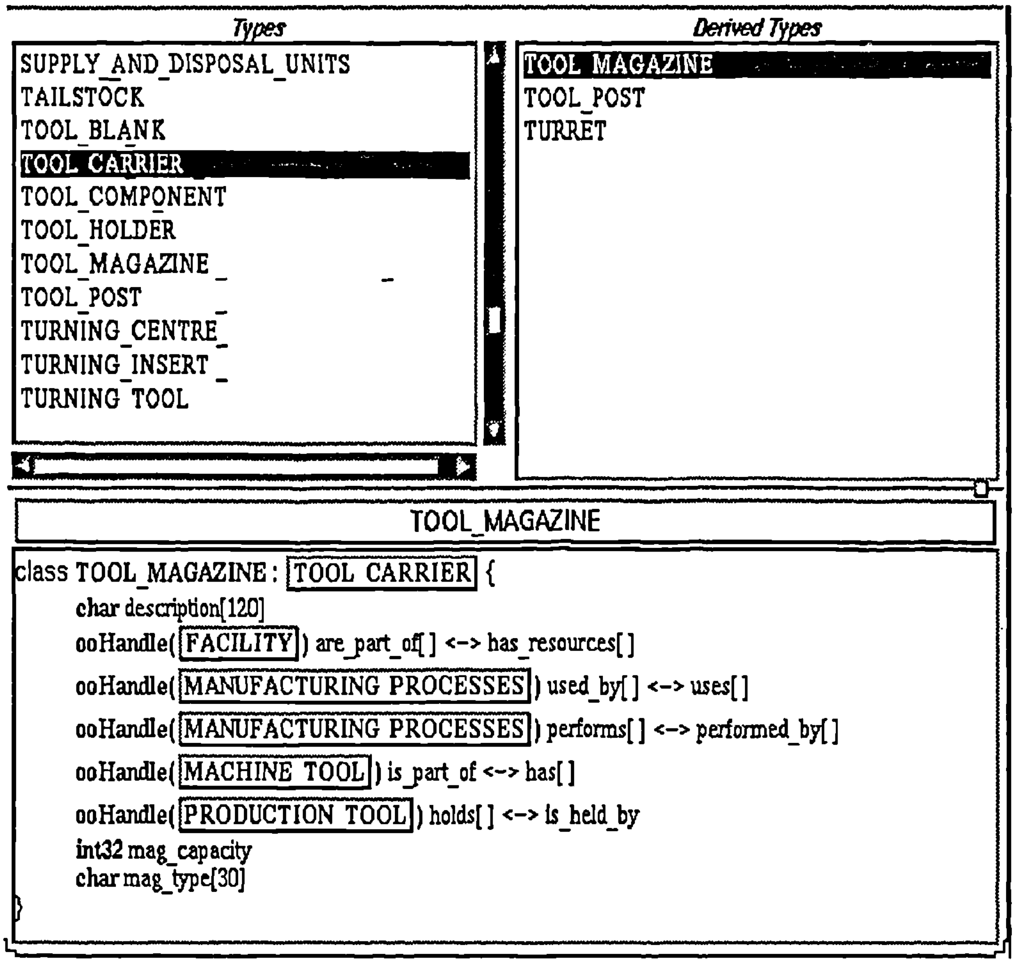


Figure H.7 Tool Magazine Schema

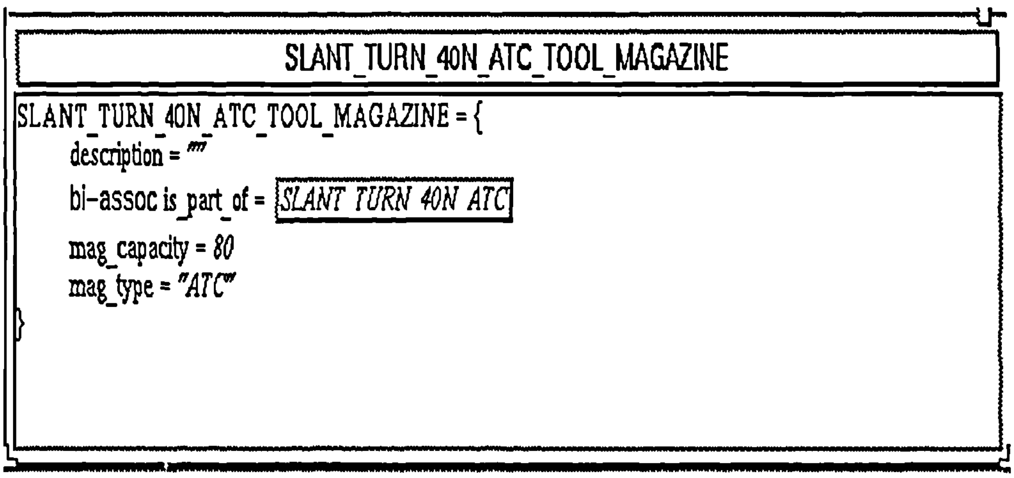


Figure H.8 Slant Turn 40N ATC Mill Centre Tool Magazine Instance

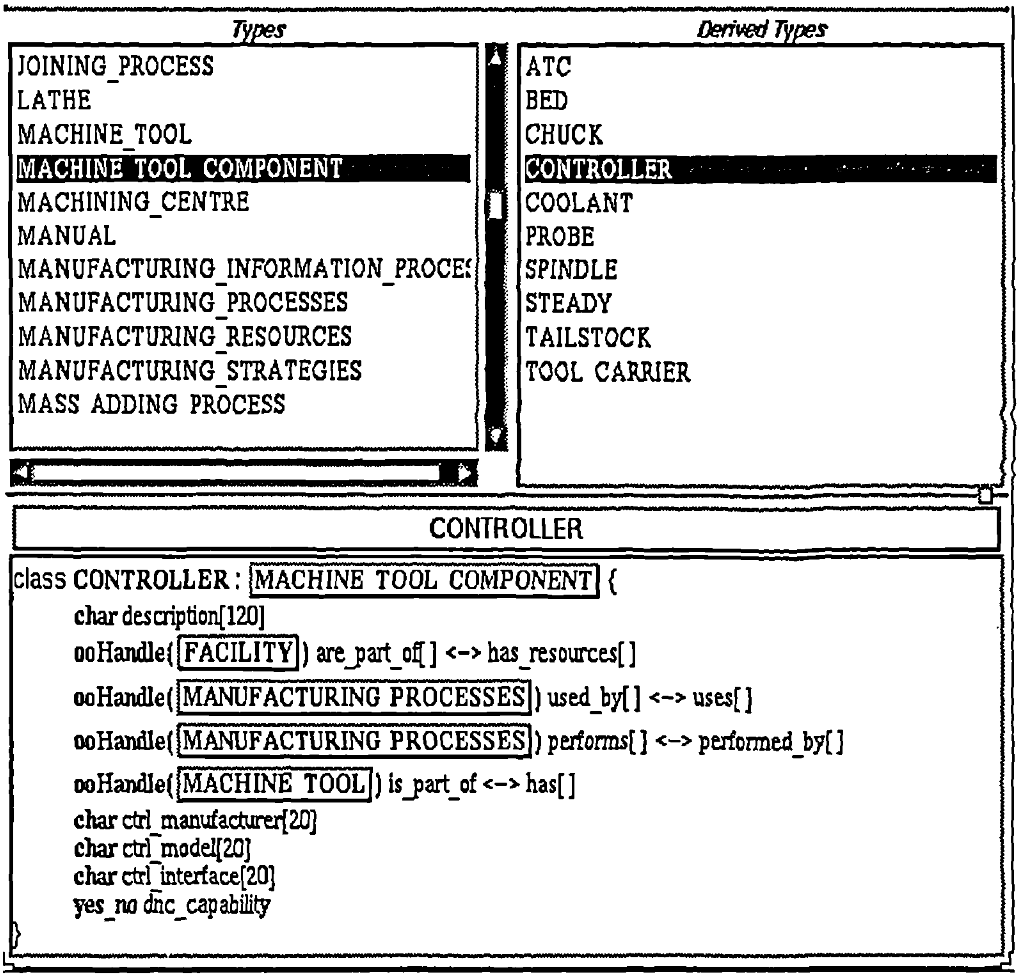


Figure H.9 Controller Schema

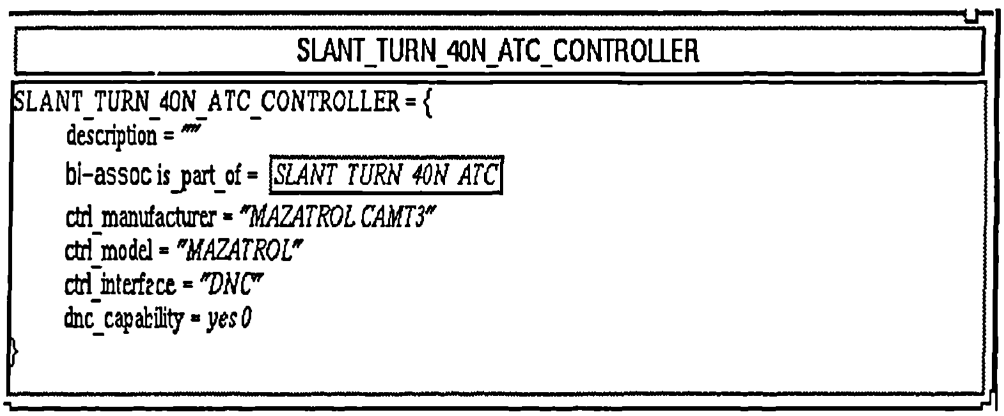


Figure H.10 Slant Turn 40N ATC Mill Centre Tool Controller Instance

```
FACE_MILL_ARBOR

FACE_MILL_ARBOR = {
    description = ""
    identification_number = 0
    company_number = "MAZAK1"
    bi-assoc is_held_by = SLANT TURN 40N ATC TURRET1
    manufacturer_code = "10"
    code = {
        tool_length_min = 16
        tool_length_max = 300
        tool_hand = 'R'
        shank_type = 'C'
        shank_width_min = 16
        shank_heigth_min = 0
        shank_width_max = 100
        shank_heigth_max = 0
    }
    bi-assoc holds = {
        FACE MILL
    }
}
```

Figure H.11 Face Mill Arbor Instance

```
MORSE_TAPER_HOLDER

MORSE_TAPER_HOLDER = {
    description = ""
    identification_number = 131072
    company_number = ""
    bi-assoc is_held_by = SLANT TURN 40N ATC TURRET1
    manufacturer_code = ""
    code = {
        tool_length_min = 16
        tool_length_max = 300
        tool_hand = 'R'
        shank_type = 'C'
        shank_width_min = 25
        shank_heigth_min = 0
        shank_width_max = 45
        shank_heigth_max = 0
    }
    bi-assoc holds = {
        DRILL
    }
}
```

Figure H.12 Morse Taper Holder Instance

```
CHUCK_HOLDER

CHUCK_HOLDER = {
    description = "\260B017"
    bi-assoc are_part_of = {
        ROTATIONAL STATION
    }
    identification_number = 0
    company_number = ""
    bi-assoc is_held_by = SLANT TURN 40N ATC TURRET1
    manufacturer_code = "MAZAK01"
    code = {
        tool_length_min = 25
        tool_length_max = 80
        tool_hand = 'R'
        shank_type = 'C'
        shank_width_min = 25
        shank_heigth_min = 0
        shank_width_max = 45
        shank_heigth_max = 0
    }
    bi-assoc holds = {
        CENTER DRILL
    }
}
```

Figure H.13 Chuck Holder Instance

```
BORING_BAR_HOLDER
BORING_BAR_HOLDER = {
    description = ""
    bi-assoc [are part of] = {
        [ROTATIONAL STATION]
    }
    identification_number = 49
    company_number = ""
    bi-assoc is_held_by = [SLANT TURN 40N ATC TURRET1]
    manufacturer_code = "MAZAK"
    code = {
        tool_length_min = 80
        tool_length_max = 400
        tool_hand = 'R'
        shank_type = 'C'
        shank_width_min = 16
        shank_heigth_min = 0
        shank_width_max = 40
        shank_heigth_max = 0
    }
    bi-assoc [holds] = {
        [BORING BAR]
        [BORING BAR WITH INSERTS]
    }
}
```

Figure H.14 Boring Bar Holder Instance

```
CUTTING_TOOL_HOLDER
CUTTING_TOOL_HOLDER = {
    description = ""
    bi-assoc [are part of] = {
        [ROTATIONAL STATION]
    }
    identification_number = 0
    company_number = "V001"
    bi-assoc is_held_by = [SLANT TURN 40N ATC TURRET1]
    manufacturer_code = "MAZAK1"
    code = {
        tool_length_min = 32
        tool_length_max = 300
        tool_hand = 'R'
        shank_type = 'S'
        shank_width_min = 25
        shank_heigth_min = 25
        shank_width_max = 25
        shank_heigth_max = 25
    }
    bi-assoc [holds] = {
        [CUTTING TOOL 25x25]
    }
}
```

Figure H.15 Cutting Tool Holder Instance

```
U_DRILL_HOLDER
U_DRILL_HOLDER = {
    description = ""
    identification_number = 0
    company_number = ""
    bi-assoc is_held_by = [SLANT TURN 40N ATC TURRET1]
    manufacturer_code = "10"
    code = {
        tool_length_min = 75
        tool_length_max = 369
        tool_hand = 'R'
        shank_type = 'C'
        shank_width_min = 20
        shank_heigth_min = 0
        shank_width_max = 40
        shank_heigth_max = 0
    }
    bi-assoc [holds] = {
        [U DRILL]
    }
}
```

Figure H.16 U Drill Holder Instance

FACE_MILL

```
FACE_MILL = {
    description = ""
    identification_number = 0
    company_number = "123"
    cutter_type = indexable_insert 1
    tool_geometry = multipoint_tool 1
    tool_movements = combination 2
    bi-assoc is_indexable_insert = [FACE MILL type]
    bi-assoc [is held by] = {
        [FACE MILL ARBOR]
    }
    mounting_type = null 0
}
```

FACE_MILL_type

```
FACE_MILL_type = {
    mounting_type = arbor 1
    bi-assoc kind_of = [FACE MILL]
    bi-assoc has = [FACE MILL insert holder]
}
```

FACE_MILL_insert_holder

```
FACE_MILL_insert_holder = {
    description = ""
    identification_number = 0
    company_number = ""
    bi-assoc part_of = [FACE MILL type]
    code_BS4193_p17 = {
        cutter_diameter = 100
        cutter_type = 'C'
        number_of_slots = 10
        cutting_direction = 'R'
        insert_holding_method = 'S'
        insert_cutting_edge_angle = 30
        insert_shape = 'L'
        insert_normal_clearance = 'D'
        indexable_insert_size = 16
        normal_clearance_wiper_edge = 'D'
        cutting_width = 8
        manufacturer_code1 = 'X'
        manufacturer_code2 = 'X'
    }
}
```

Figure H.17 Indexable Insert Face Mill instance composed by three instances Face Mill, Face Mill Instance and Face Mill Insert Holder

```
DRILL

DRILL = {
    description = ""
    identification_number = 0
    company_number = "351778005000"
    cutter_type = solid_cutter 0
    tool_geometry = multipoint_tool 1
    tool_movements = translation 0
    bi-assoc is_solid_cutter = DRILL_type
    bi-assoc is held by = {
        MORSE TAPER HOLDER
    }
}
```

```
DRILL_type

DRILL_type = {
    bi-assoc kind_of = DRILL
    code_BS328 = {
        tool_diameter = 16
        tool_type = 'S'
        tool_length = 218
        flute_length = 120
        lead_of_helix = 120
        tool_hand = 'R'
        shank_type = 'E'
        shank_size = 1
        manufacturer_code1 = 'X'
        manufacturer_code2 = 'X'
    }
}
```

Figure H.18 Solid Drill Instance composed by two instances Drill and Drill Type

```
BORING_BAR

BORING_BAR = {
    description = ""
    bi-assoc [are part of] = {
        [ROTATIONAL STATION]
    }
    identification_number = 0
    company_number = "SANDVICK"
    cutter_type = solid_cutter 0
    tool_geometry = single_point_tool 0
    tool_movements = combination 2
    bi-assoc is_solid_cutter = [BORING BAR type]
    bi-assoc [is held by] = {
        [BORING BAR HOLDER]
    }
}
```

```
BORING_BAR_type

BORING_BAR_type = {
    bi-assoc kind_of = [BORING BAR]
    code_BS1296_p3 = {
        tool_type = 'S'
        tool_shape = 50
        tool_hand = 'R'
        tool_length = 200
        shank_type = 'C'
        shank_width = 25
        shank_height = 0
        manufacturer_code1 = 'X'
        manufacturer_code2 = 'X'
        manufacturer_code3 = 'X'
    }
}
```

Figure H.19 Solid Boring Bar Instance composed by two instances Boring Bar and Boring Bar Type

```
CUTTING_TOOL_25x25

CUTTING_TOOL_25x25 = {
    description = ""
    bi-assoc [are part of] = {
        [ROTATIONAL STATION]
    }
    identification_number = 2
    company_number = "SANDVICK"
    cutter_type = indexable_insert 1
    tool_geometry = single_point_tool 0
    tool_movements = translation 0
    bi-assoc is_indexable_insert = [CUTTING TOOL 25x25 type]
    bi-assoc [is held by] = {
        [CUTTING TOOL HOLDER]
    }
}
```

```
CUTTING_TOOL_25x25_type

CUTTING_TOOL_25x25_type = {
    mounting_type = null 0
    bi-assoc kind_of = [CUTTING TOOL 25x25]
    bi-assoc has = [CUTTING TOOL 25x25 insert holder]
}
```

```
CUTTING_TOOL_25x25_insert_holder

CUTTING_TOOL_25x25_insert_holder = {
    description = ""
    identification_number = 0
    company_number = ""
    bi-assoc part_of = [CUTTING TOOL 25x25 type]
    code_BS4193_p6 = {
        insert_holding_method = 'M'
        insert_shape = 'L'
        tool_style = 'A'
        insert_normal_clearance = 'A'
        tool_hand = 'R'
        tool_heigth = 25
        shank_width_or_cartridge_type = 5
        tool_length = 'H'
        indexable_insert_size = 16
        special_tolerances = 'Q'
        manufacturer_code1 = 'X'
        manufacturer_code2 = 'X'
        manufacturer_code3 = 'X'
    }
}
```

Figure H.20 Cutting Tool 25x25 Instance composed by three instances

```
U_DRILL

U_DRILL = {
    description = ""
    identification_number = 0
    company_number = "10"
    cutter_type = indexable_insert 1
    tool_geometry = multipoint_tool 1
    tool_movements = translation 0
    bi-assoc is_indexable_insert = [U DRILL type]
    bi-assoc [is held by] = {
        [U DRILL HOLDER]
    }
}
```

```
U_DRILL_type

U_DRILL_type = {
    mounting_type = null 0
    bi-assoc kind_of = [U DRILL]
    bi-assoc has = [U DRILL insert holder]
}
```

```
U_DRILL_insert_holder

U_DRILL_insert_holder = {
    description = ""
    identification_number = 0
    company_number = ""
    bi-assoc part_of = [U DRILL type]
    code_sandvik = {
        tool_diameter = 20
        tool_type = 'B'
        tool_length = 344
        flute_length = 195
        lead_of_helix = 195
        tool_hand = 'R'
        shank_type = 'A'
        shank_size = 100
        number_of_slots = 2
        insert_holding_method = 'P'
        insert_shape = 'L'
        insert_normal_clearance = 'C'
        manufacturer_code1 = 'X'
        manufacturer_code2 = 'X'
    }
}
```

Figure H.21 U Drill Instance composed by three instances

TURNING

```
class TURNING : CHIP_FORMING_PROCESS, MECHANICAL_PROCESS, SOLID_MATERIAL_PROCESS {
    char description[120]
    float32 cost
    char mpdescription[120]
    ooHandle(FACILITY) are_part_of <-> has_processes[]
    ooHandle(MANUFACTURING_RESOURCES) uses[] <-> used_by
    ooHandle(MANUFACTURING_RESOURCES) performed_by <-> performs[]
    float32 tolerance_minimum
    float32 tolerance_maximum
    float32 surface_finish_min
    float32 surface_finish_max
    typeform form
    ooHandle(WORKPIECE) requires_workpiece <-> is_required_by
    ooHandle(TOOL) requires_tool[] <-> is_required_by
    char description[120]
    ooVArray(STRING) pre_process
    ooVArray(STRING) post_process
    char description[120]
    state state_of_material
    ooVArray(material) type_of_material
}
```

OD_TURNING_TURN_PROC

```
OD_TURNING_TURN_PROC = {
    description = ""
    cost = 0
    mpdescription = "External turning process for outside diameters"
    tolerance_minimum = 0
    tolerance_maximum = 0
    surface_finish_min = 0
    surface_finish_max = 0
    form = null_form 0
    bi-assoc requires_workpiece = #2-2-3-29
    bi-assoc requires_tool = {
        #2-2-3-32
    }
    description = ""
    pre_process = {
    }
    post_process = {
    }
    description = ""
    state_of_material = solid 1
    type_of_material = {
    0   [0] ferrous_metals 1
        [1] null_material 0
        [2] null_material 0
        [3] null_material 0
        [4] null_material 0
        [5] null_material 0
```

Figure H.22 OD Turning Instance

```
#2-2-3-29

WORKPIECE #2-2-3-29 = {
    int32 number_of_axis = 1
    ooVArray(axis_of_motion) motions_available = {
    0   {
            motion_in = rz_axis 7
            extent_motion = 0
            minimum_increment = 0
            power = 0
            torque = 0
        }
    }
    typeform form = rotational 1
}
```

```
#2-2-3-32

TOOL #2-2-3-32 = {
    int32 number_of_axis = 2
    ooVArray(axis_of_motion) motions_available = {
    0   {
            motion_in = x_axis 1
            extent_motion = 0
            minimum_increment = 0
            power = 0
            torque = 0
        }
    }
    tool_name name_of_tool = external 0
    tool_type hand_of_tool = neutral 1
    tool_direction direction_of_feed = forward 0
    float32 nose_radio = 0
    int32 allowable_depth = 0
    int32 included_angle_rough = 0
    int32 included_angle_finish = 0
    tool_position position_of_the_tool = perpendicular 1
    tool_feed tool_is = statical 1
    ooHandle(PRODUCTION_TOOL) type_of <-> supertype_of = CUTTING TOOL 25x25
}
```

Figure H.23 Workpiece and Tool Instances
of the OD Turning Process Instance

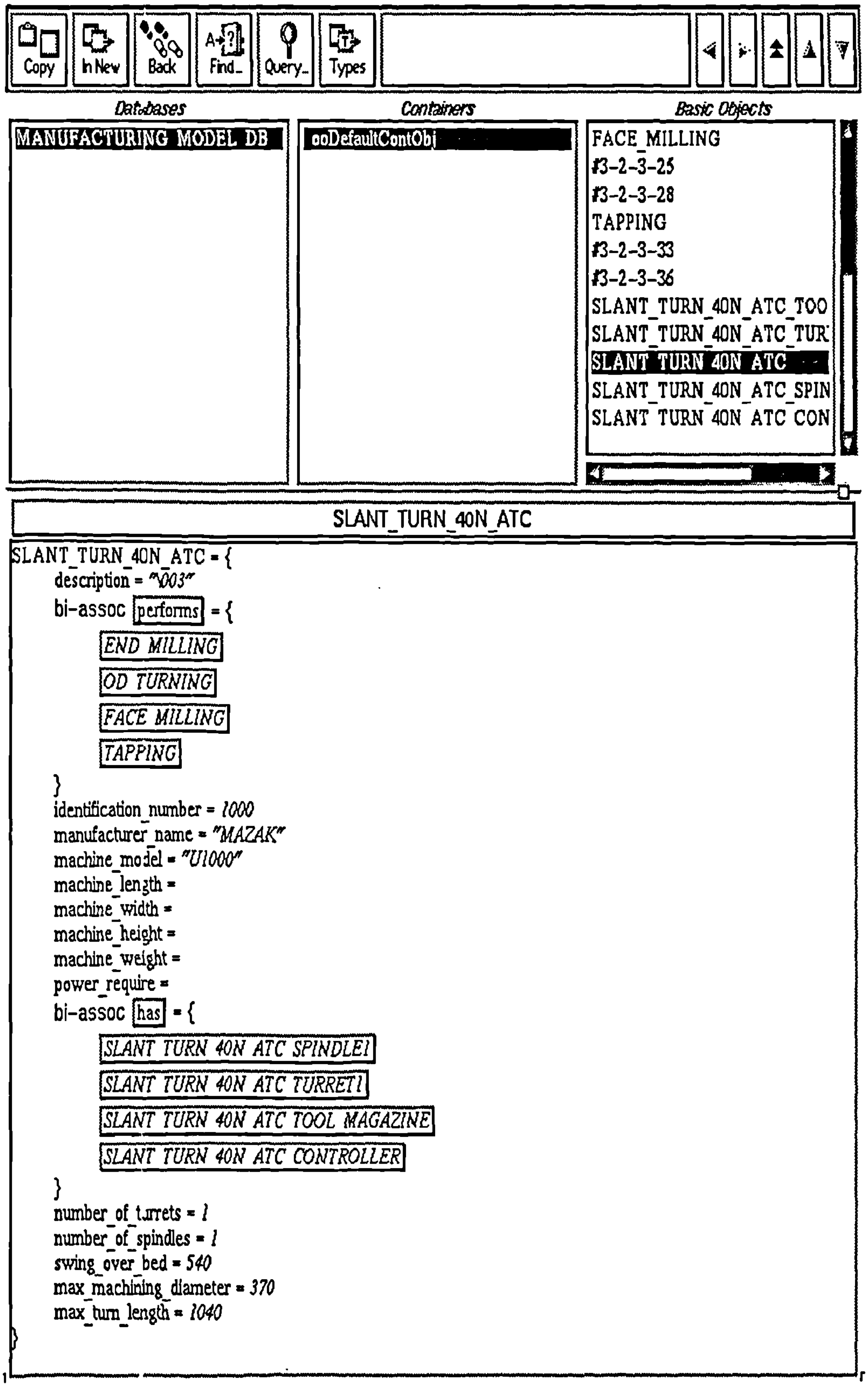


Figure H.24 Slant Turn 40N ATC Mill Centre process capabilities

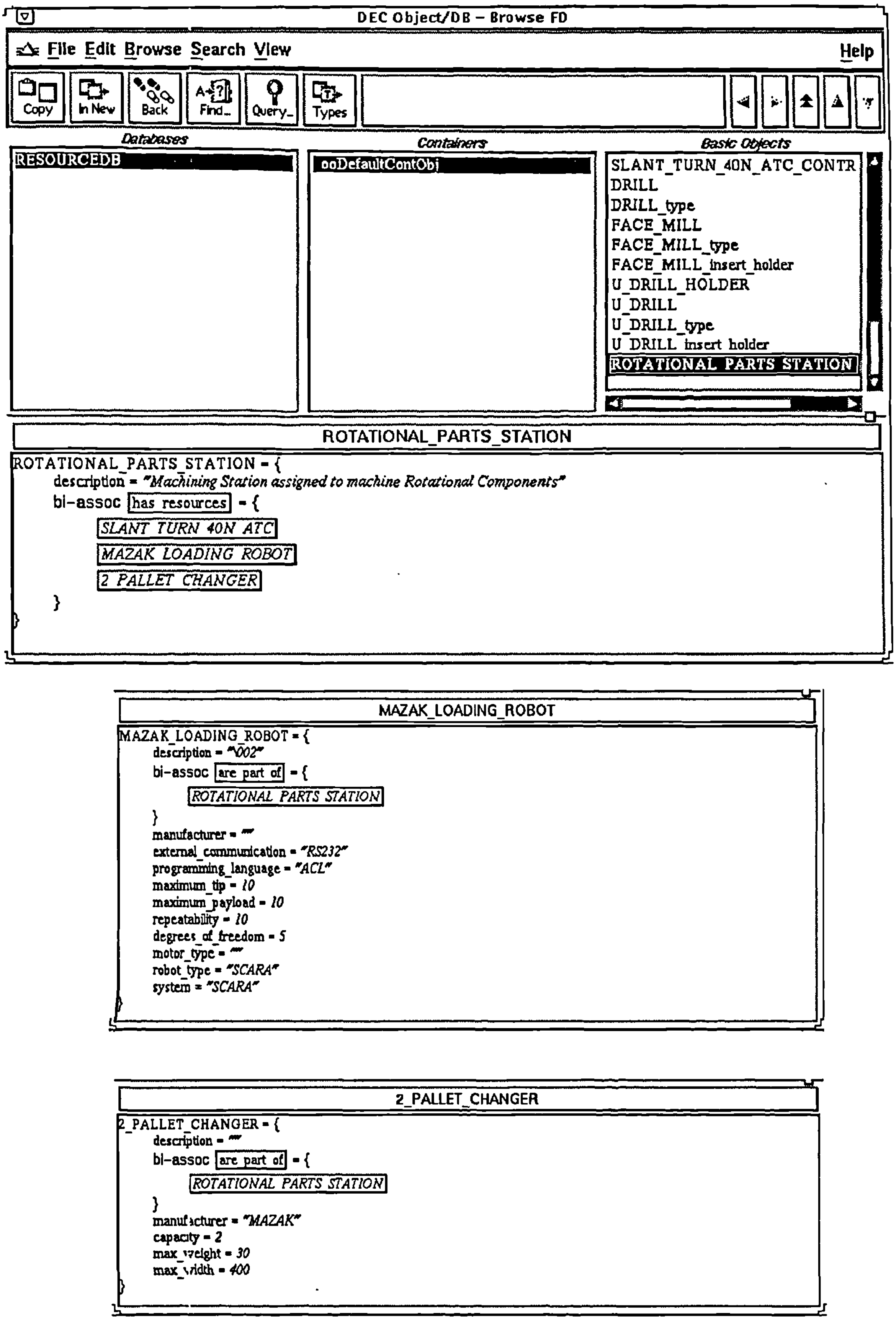


Figure H.25 Rotational Parts Station Configuration

Appendix I

Instances of the Rotational Parts Line

I.1. Introduction

This appendix presents the instances of the Manufacturing Model database which represent the Rotational Parts Line and Assembly Line described in Chapter 15.

I.2. Factory Layout Instances

The YAMAZAKI WORCESTER class illustrated in figure I.1 represents the partial model described in Chapter 15, Section 15.3 (Figure 15.2). The YAMAZAKI WORCESTER *includes* MACHINING LINE, SUPERFINISHING AND QUALITY CONTROL, SHEET METAL WORKING, ASSEMBLY, PRECISION ASSEMBLY, PAINTING, AUTOMATED WAREHOUSE and TRANSPORTER AGV instances. Only the MACHINING LINE has been decomposed further and *includes* LARGE PRISMATIC MACHINING LINE, SMALL PRISMATIC MACHINING LINE and ROTATIONAL PARTS LINE (Figure I.2).

I.3. Rotational Parts Line Instance

The instance of the Rotational Parts Line is shown in figure I.3, and *has* two resources AUTO STACKING CRANE and PALLET STOCKER. It also *includes* three ROTATIONAL PARTS STATIONS, this is accordingly to the model presented in figure 15.4. The AUTO STACKING CRANE process capabilities are illustrated in figure I.4. The instance of the PALLET STOCKER is in figure I.5

I.4. Assembly Instance

Figure I.6 shows the Assembly instance and this instance includes SUBASSEMBLY, ASSEMBLY LINE, INSPECTION and SHIPPING. Further more, the ASSEMBLY LINE is composed of three sections: A LINE, GENERAL ASSEMBLY, ELECTRICAL ASSEMBLY and MECHANICAL ADJUSTMENT (Figure I.7).

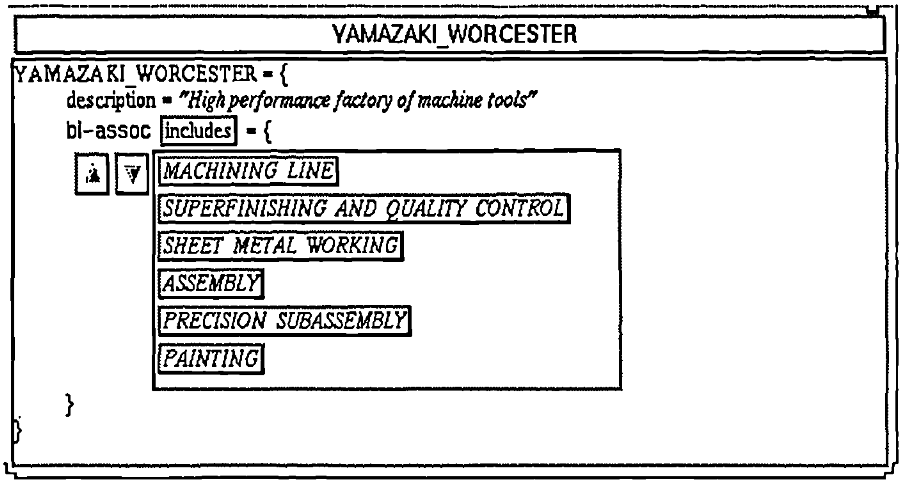


Figure I.1 Mazak European Factory Instance

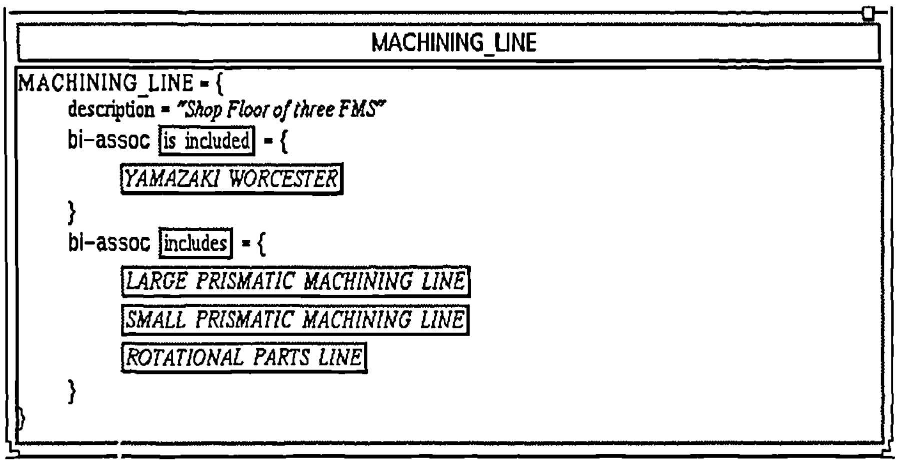


Figure I.2 Machining Line Instance

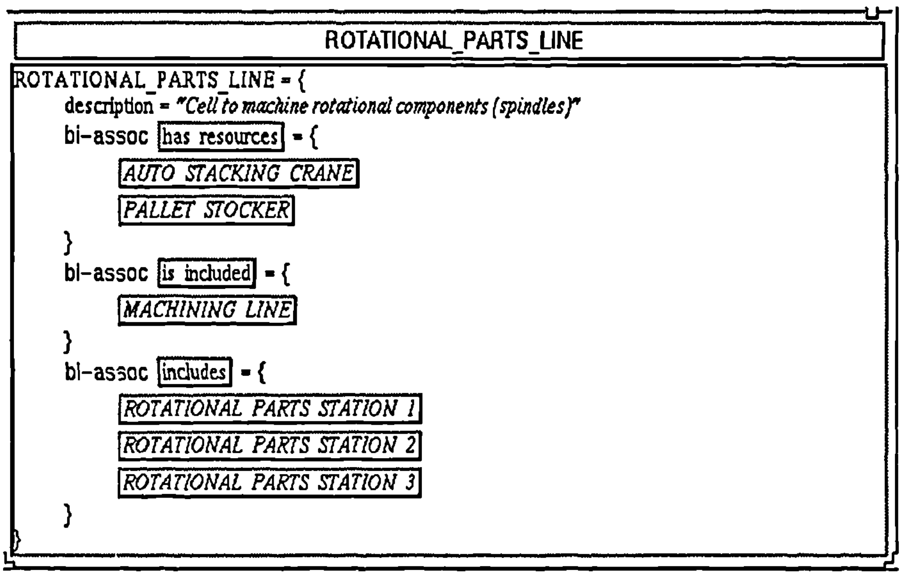


Figure I.3 Rotational Parts Line Instance

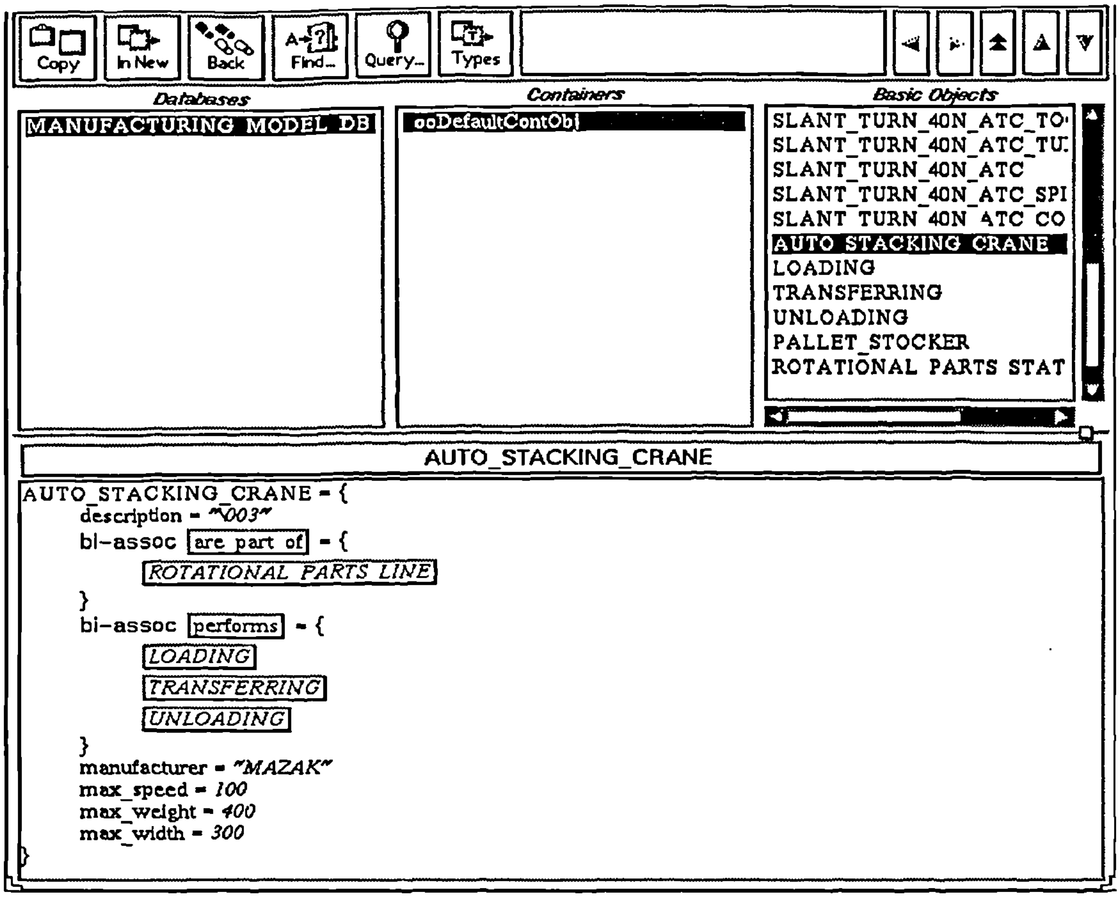


Figure I.4 Auto Stacking Crane Process Capabilities

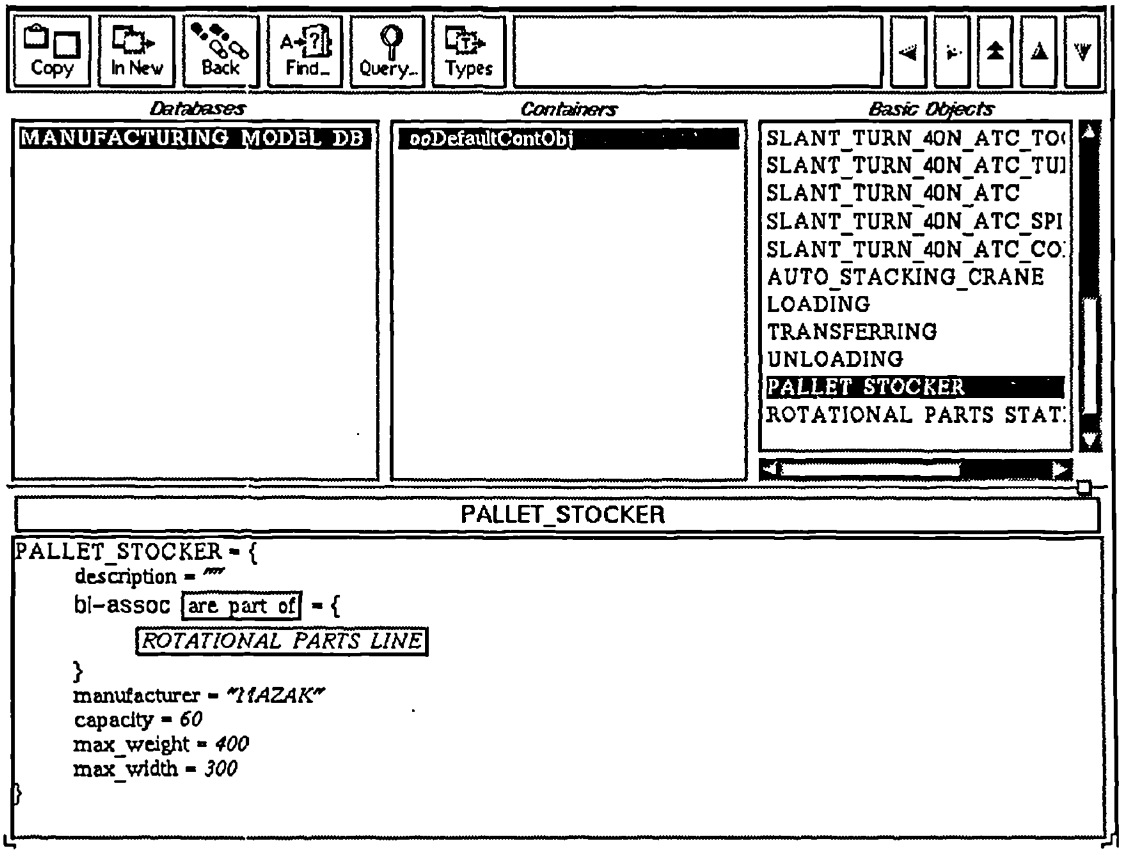


Figure I.5 60 Pallet Stocker Instance

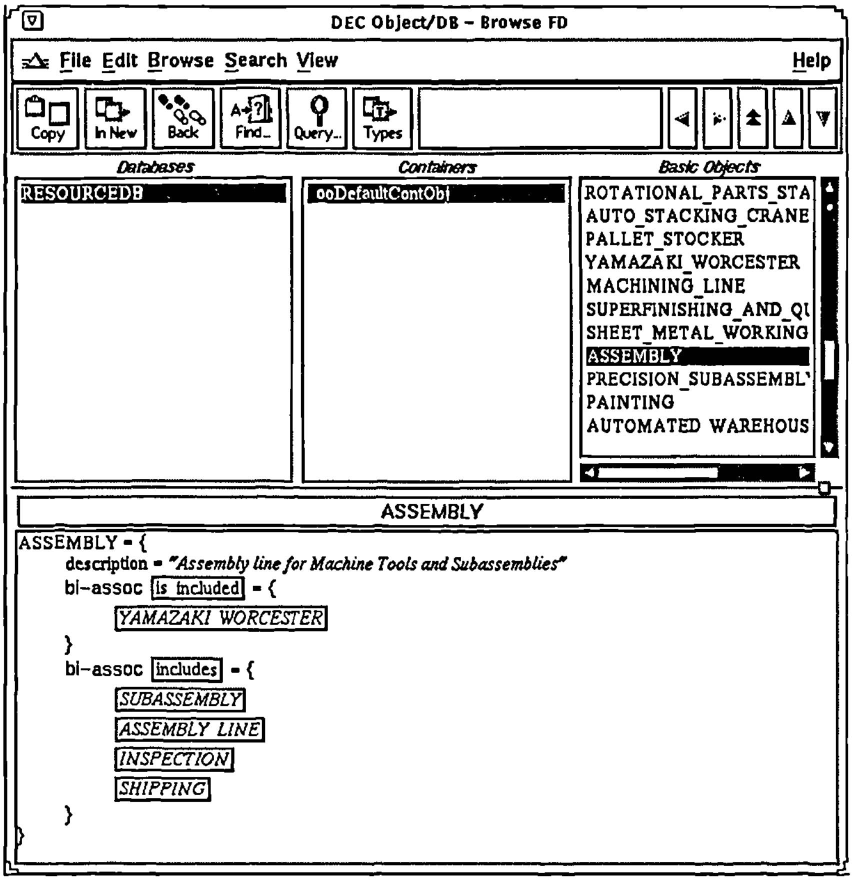


Figure I.6 Assembly Instance

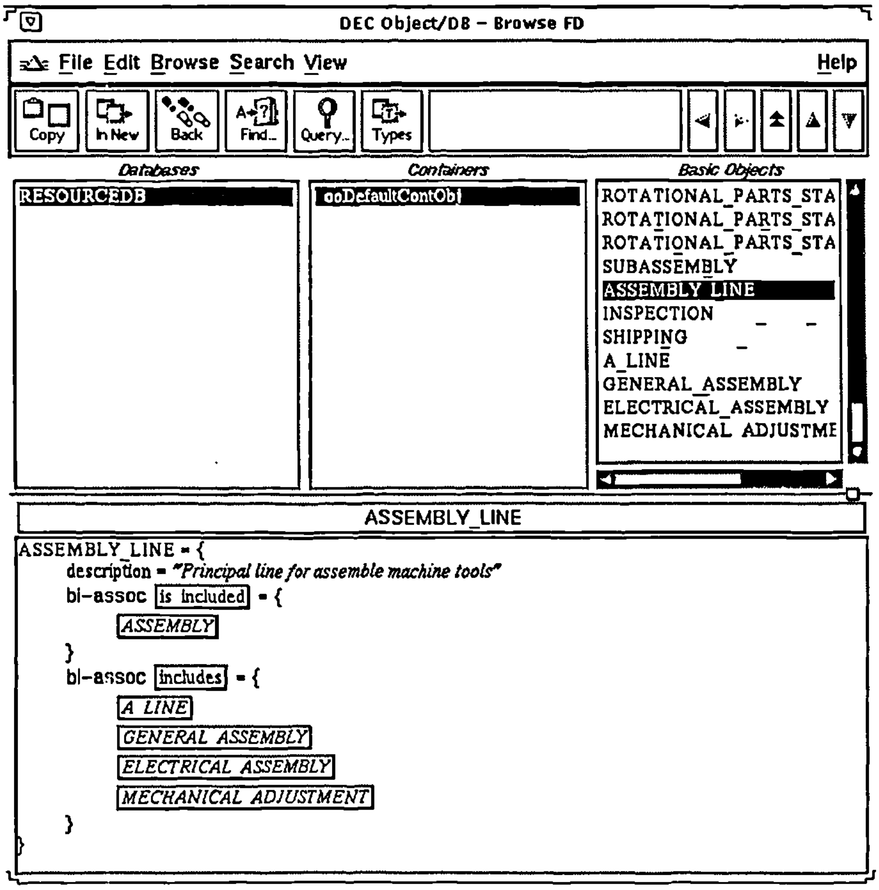


Figure I.7 Assembly Line Instance

Appendix J

Manufacturing Model Design and Implementation

J.1. Introduction

This appendix describes the design and implementation of the Manufacturing Model. The design conforms the Computation Viewpoint of the CAE–RM. The important computational objects of the Manufacturing Model have been defined to be the following: Information Model, Information Model Manager and Information Model Interface. A session of the Manufacturing Model Interface is presented to the reader to highlight its functionality.

J.2. Computation Viewpoint of the Manufacturing Model

The computational objects of the Manufacturing Model are three: Information Model, Information Model Manager and Information Model Interface (See Chapter 9, figure 9.2). These computational objects describes the Manufacturing Model in terms of the operations, interactions and functions. In order to be more specific these objects have been renamed to: Manufacturing Model Database, Manufacturing Model Manager and Manufacturing Model Interface (Figure J.1).

J.2.1. Manufacturing Model Database

The Manufacturing Model Database (MMD) is the object which the represents the functionality of the software which allows the Manufacturing Model to represent the manufacturing resources, processes, strategies and facilities, i.e. the object oriented definitions on the object oriented database.

J.2.2. Manufacturing Model Manager

The Information Model Manager (MMM) is the object which represents the functionality of the software which allows the Manufacturing Model to be populated

J.2.3. Manufacturing Model Interface

The Information Model Interface (MMI) is the object which represent the functionality of the software which allows the users to query and access the information in the Manufacturing Model Database.

J.2.4. Relations between the Information Model, Information Model Manager and Information Model Interface

The MMD has to be accessed by two type of users, therefore two relations are defined between the MMD and the Manufacturing Model Manager (MMM) and Manufacturing Model Interface (MMI). The relation with the MMM allows the user to create the instances which describe a particular Manufacturing Model, the relation with the MMI enables user to query and derive information from the instantiated Manufacturing Model (figure J.1)

J.3. Manufacturing Model Implementation

This Manufacturing Model has been implemented using the Object Oriented Database Object DEC/DB and the Object Oriented Language C++.

J.3.1. Implementation of the Manufacturing Model Database

The implementation of the Information Model is more a less straightforward. The EXPRESS model has to be translated into the Data Definition Model of the Object DEC/DB. Modifications have to be made in special cases due to the limitations of the databases, for example there is only a limited support for Multiple–inheritance and the different EXPRESS constructs (e.g. SELECT), therefore these constructs have to be implemented using the methods provided by Object DEC/DB.

J.3.2. Implementation of the Manufacturing Model Manager

The Manufacturing Model Manager is the program which allows the user to populated the Manufacturing Model Database. This manager allows the user to:

1. Create types of Resources, Process and Strategies

2. Create definitions of Stations, Cells, Shop Floors and Factories

3. Update the Manufacturing Model Database

4. Verify the contents of the Manufacturing Model Database

All these functions are implemented in C++ and its functionality is easily extendible.

J.3.3. Implementation of the Manufacturing Model Interfaces

There are two Information Model Interfaces: one is provide directly by the software tool used, i.e. the "ootoolmgr" of DEC Object DB, and the second one is a program implemented by the author based on a set of functions programmed in C++ and remote procedures calls. which allows to have access to the information held in the Manufacturing Model Database.

The "ootoolmgr" of DEC Object DB is a software which allow the user to verify the schema developed and interrogate the database (Figure J.2). It also provide the means to follow the association among the object instantiated. This tool was found very useful during the development of the Manufacturing Model prototype because the verification of the relations among the manufacturing entities was possible.

The second interface was developed by the author to interact with the Manufacturing Model Database in a more user friendly manner. Examples of this interface are presented in figures J.3 through J.12.

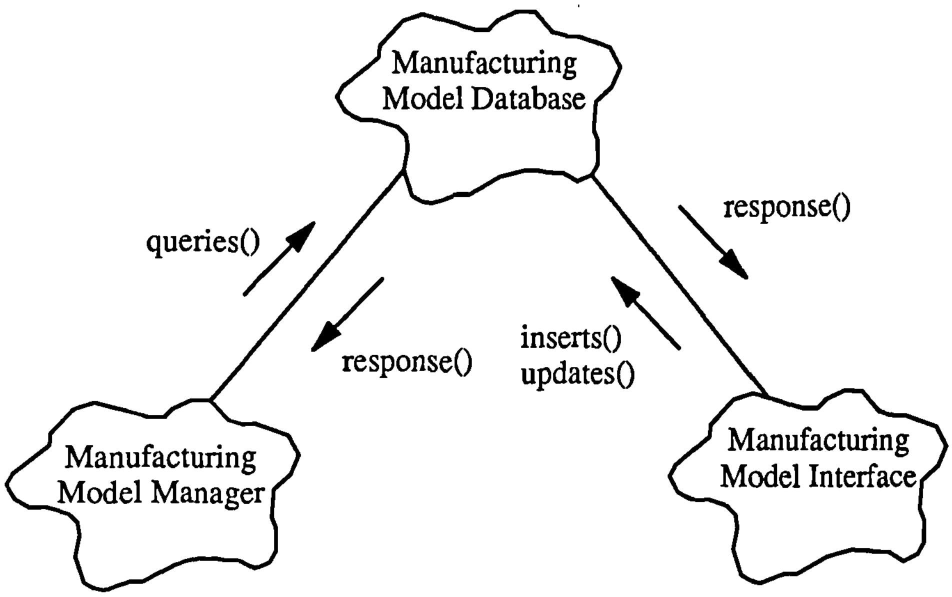


Figure J.1 Partial Computational Viewpoint of the Manufacturing Model

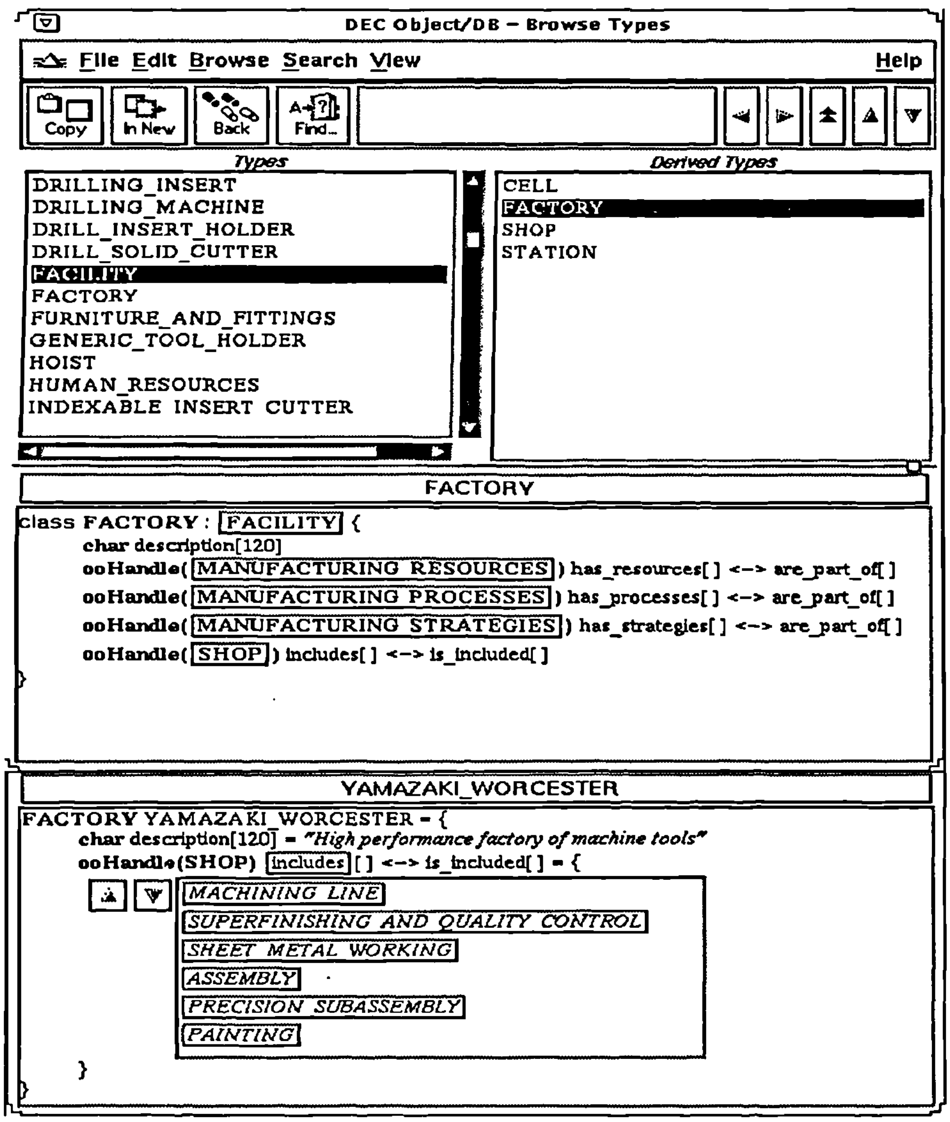


Figure J.2 Example of the ootoolmgr of DEC Object/DB

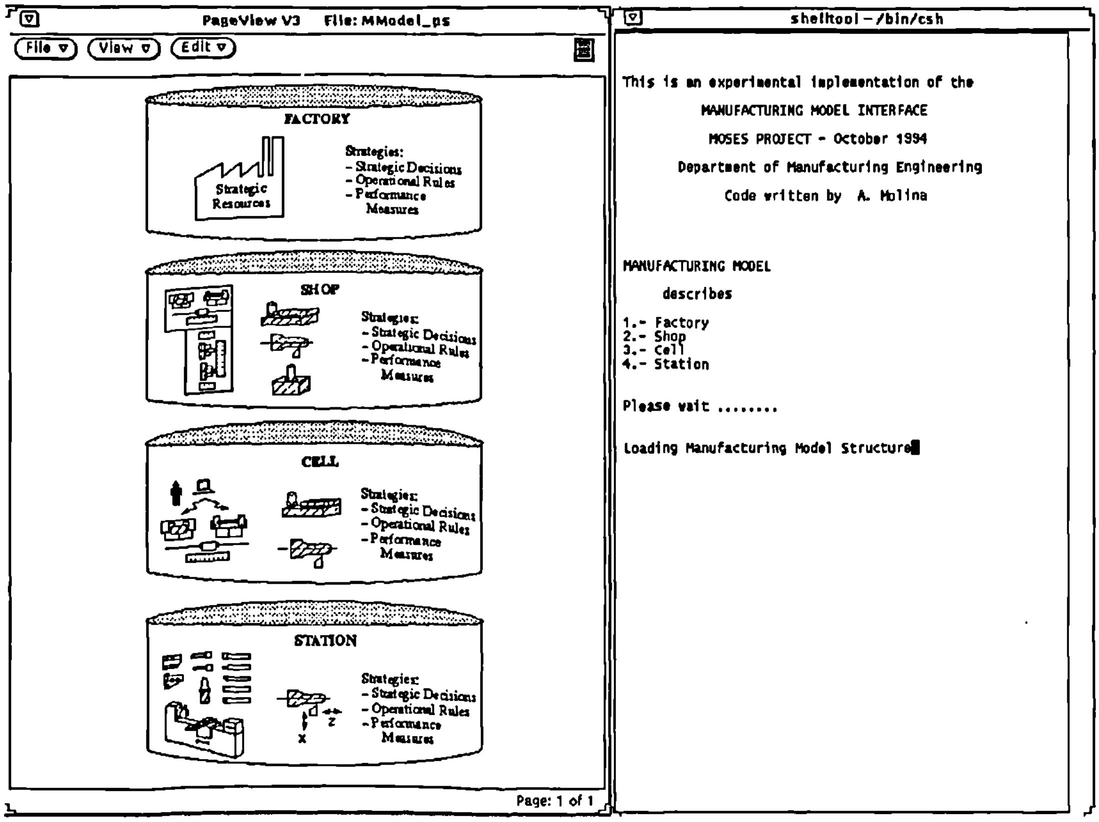


Figure J.3 Manufacturing Model Interface Menu

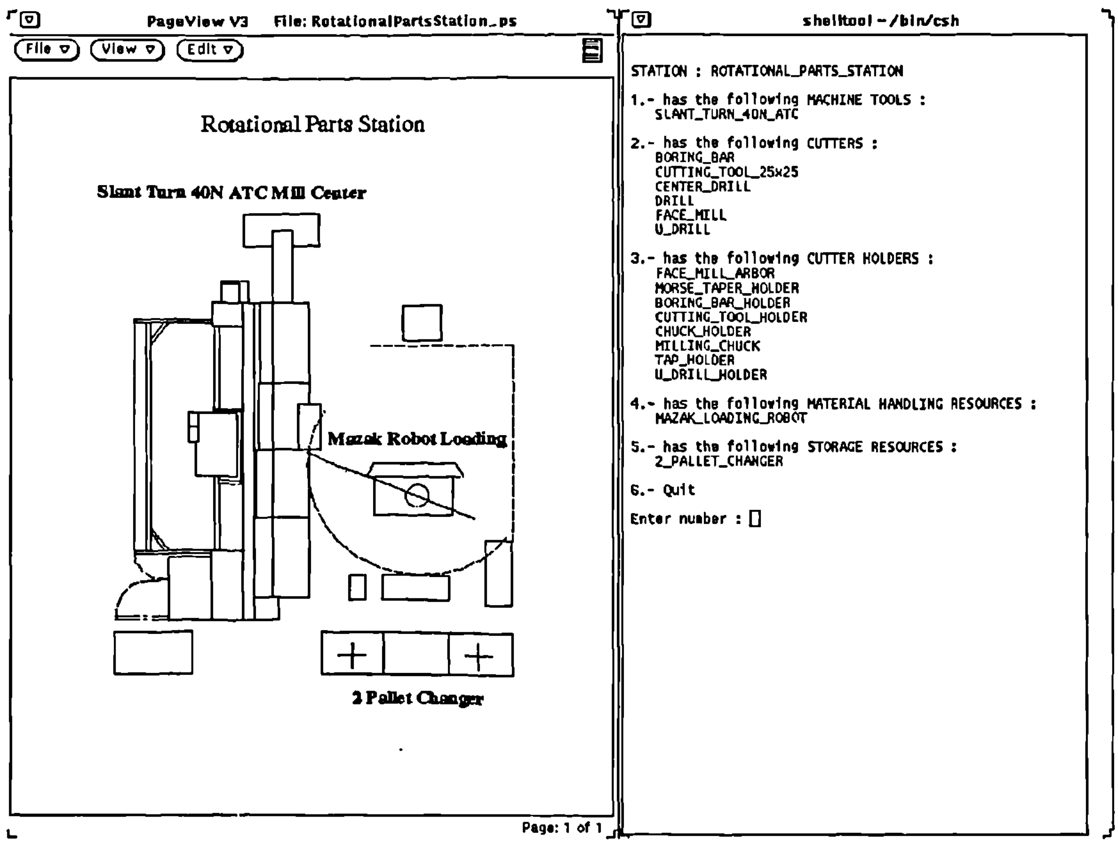


Figure J.4 Rotational Parts Station Configuration

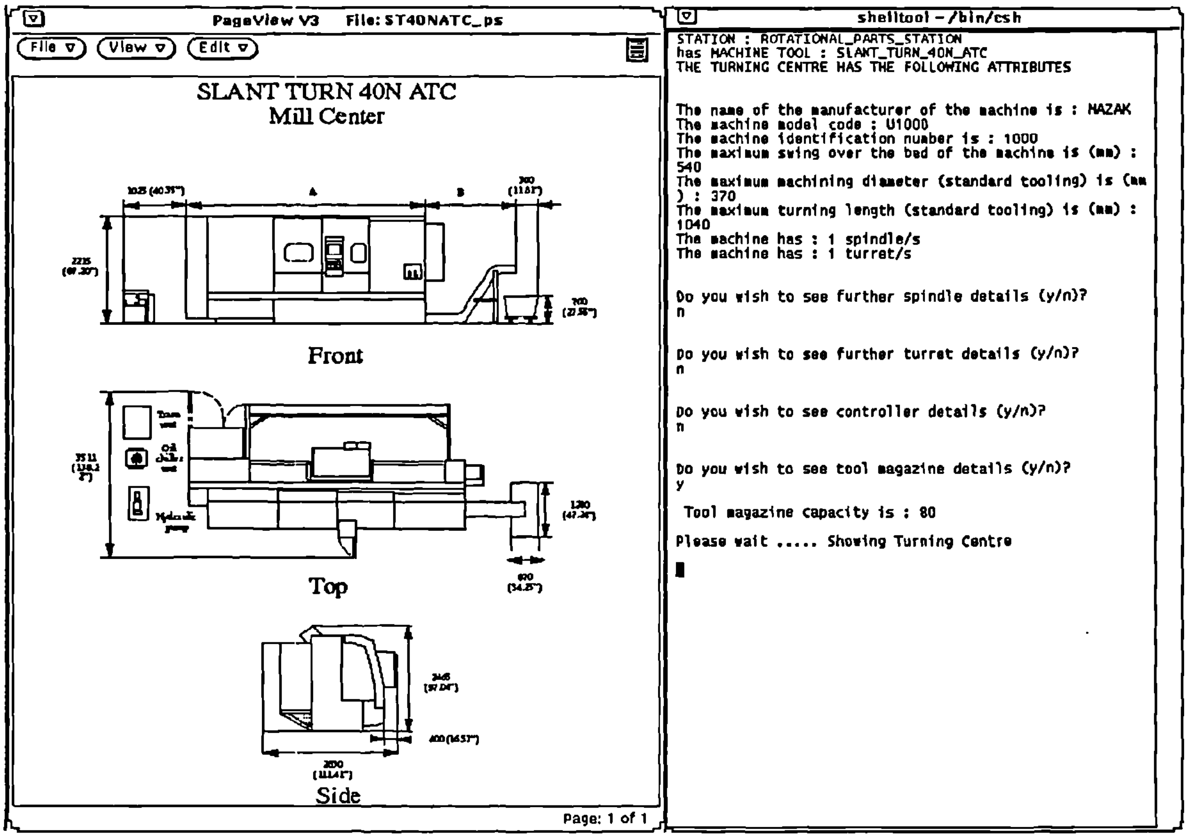


Figure J.5 Slant Turn 40N ATC Description

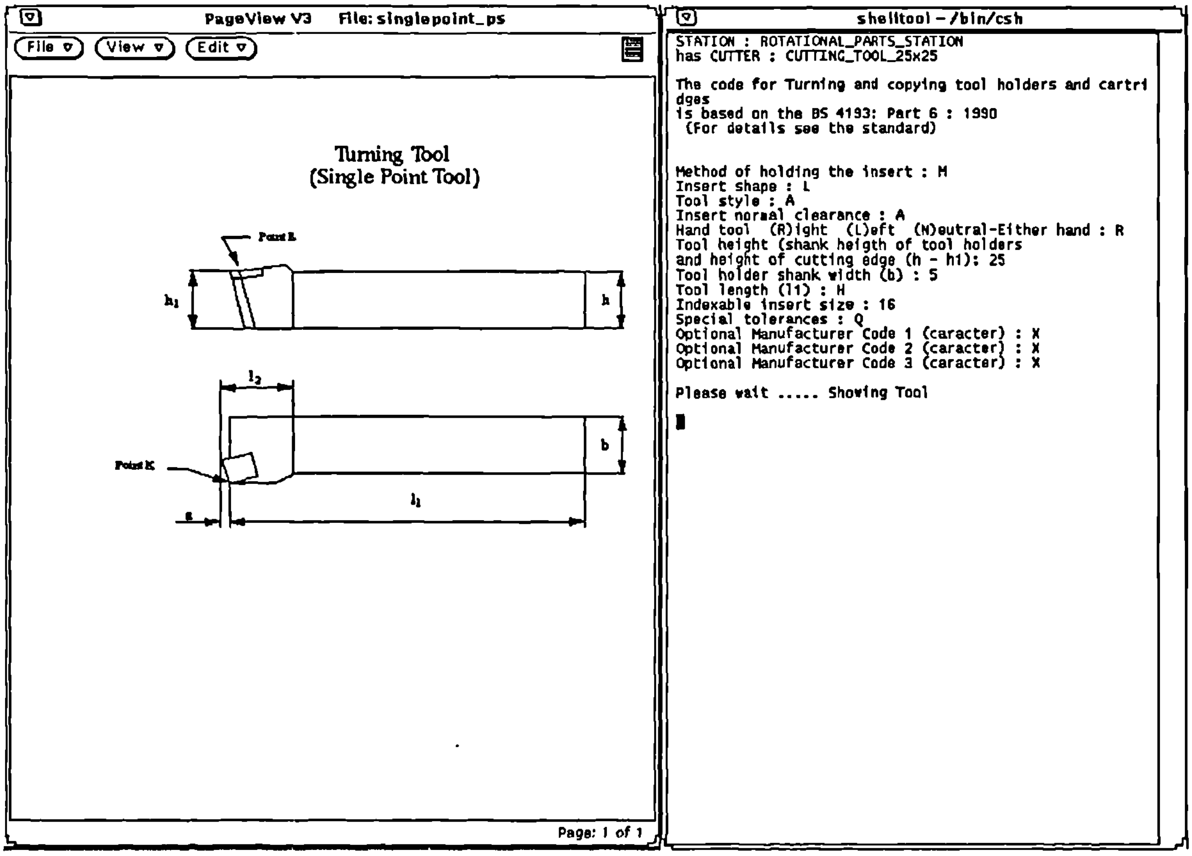


Figure J.6 Turning Tool 25 x 25 Description

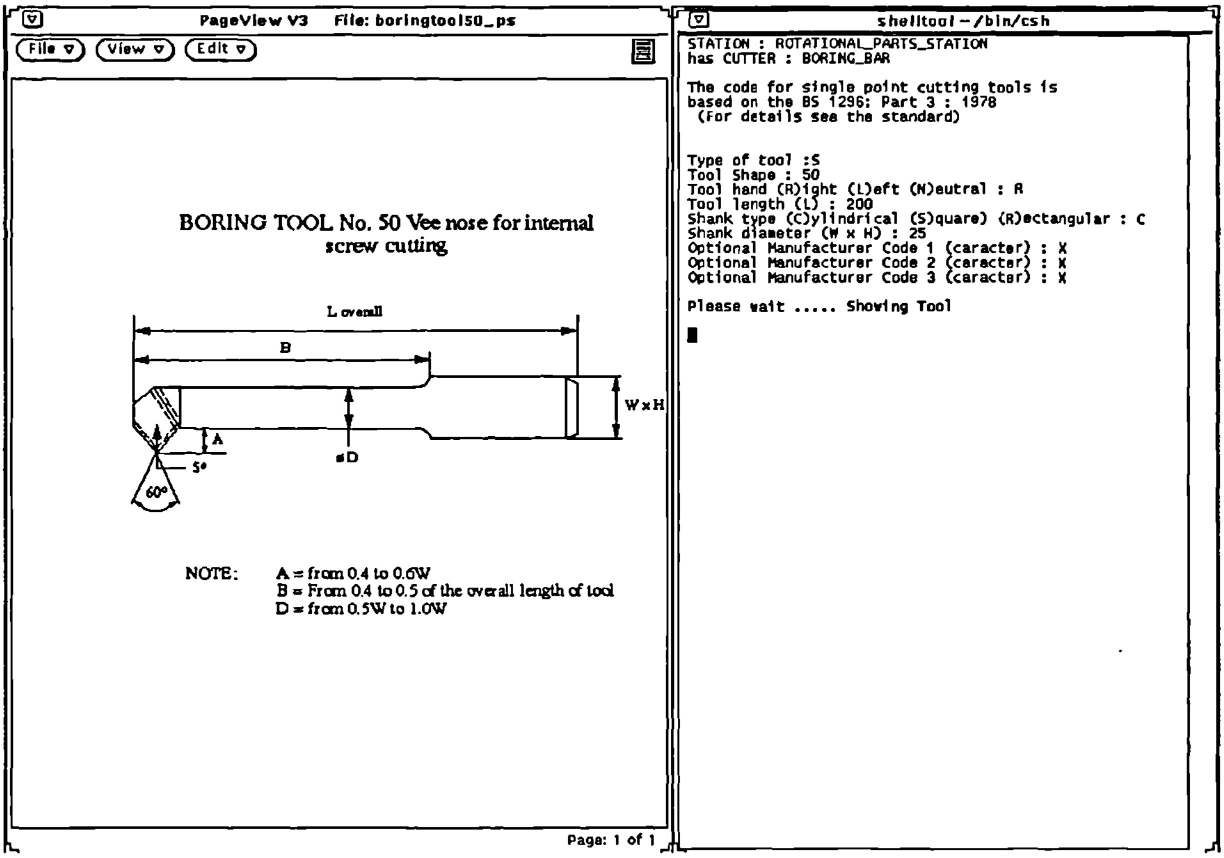


Figure J.7 Boring Tool Description

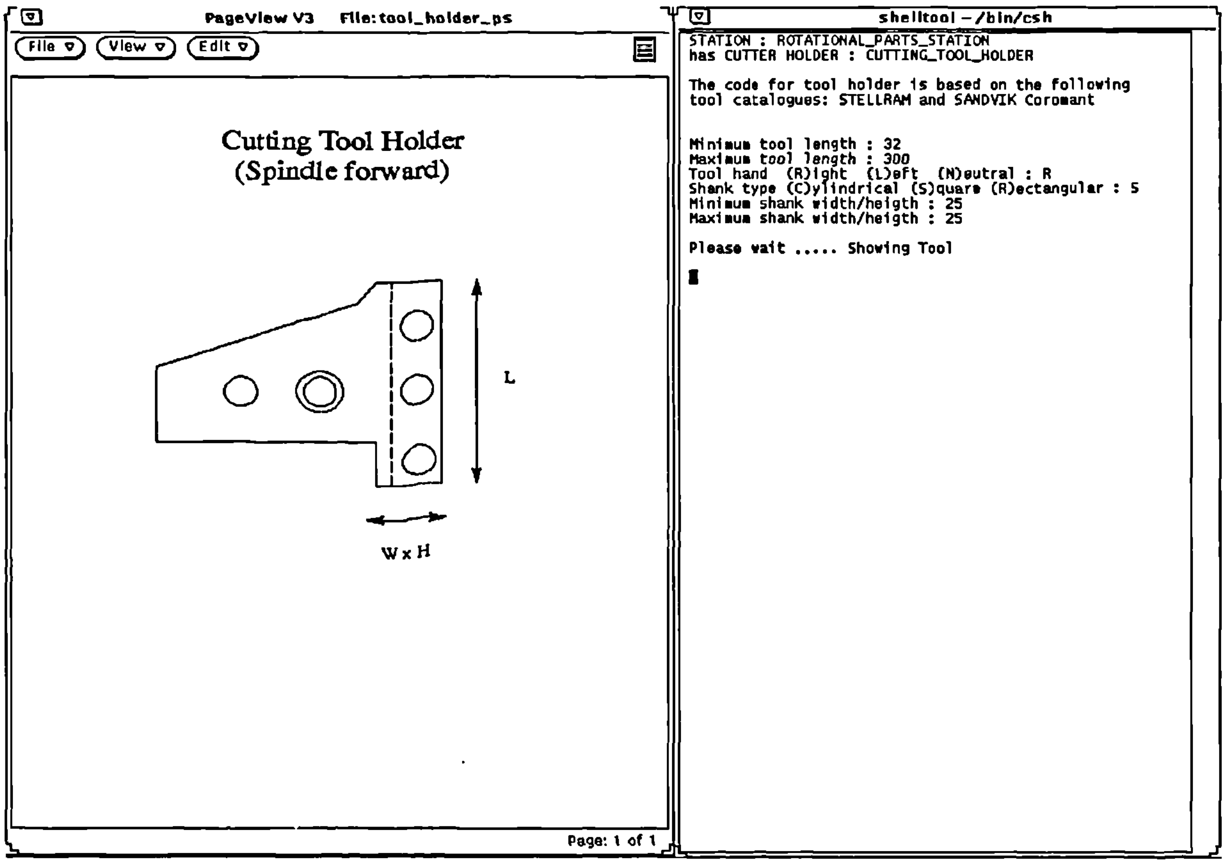


Figure J.8 Cutting Tool Holder Description

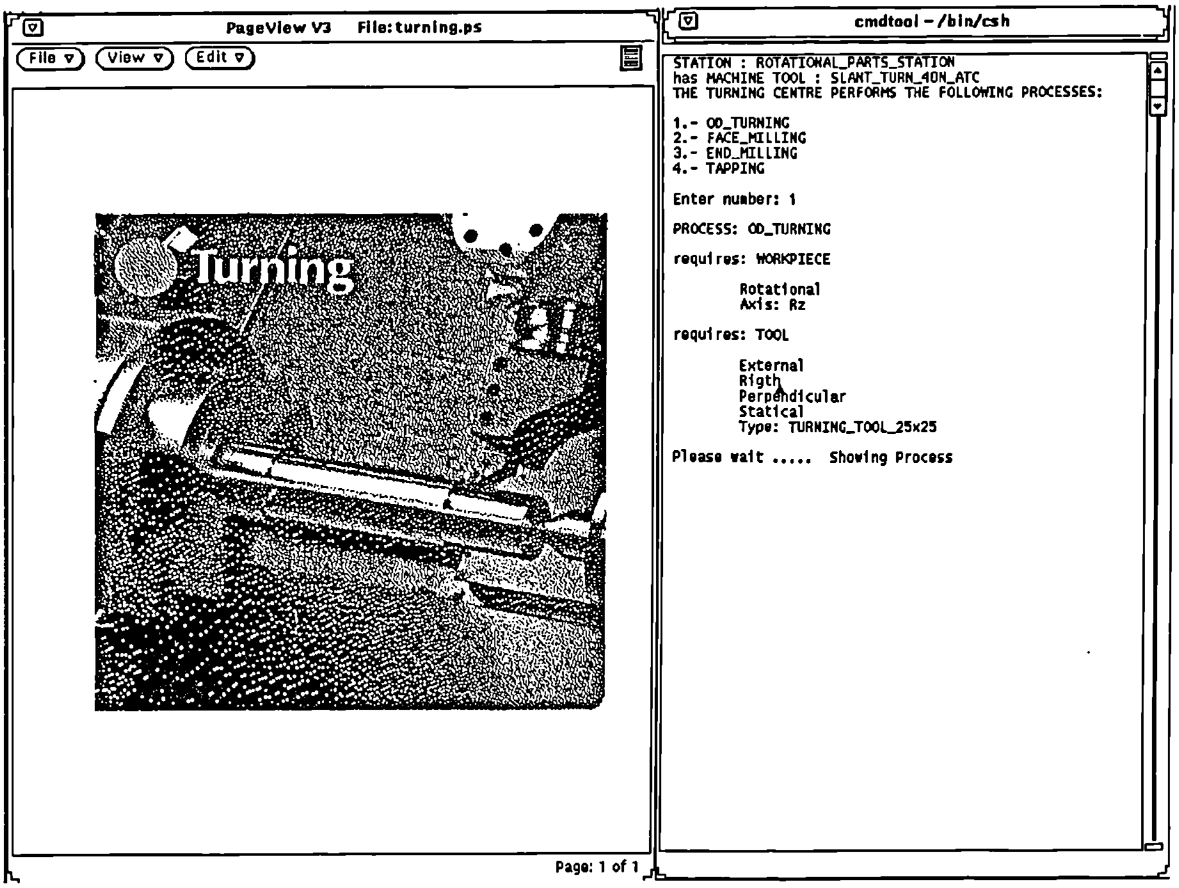


Figure J.9 Turning Process Capabilities Description

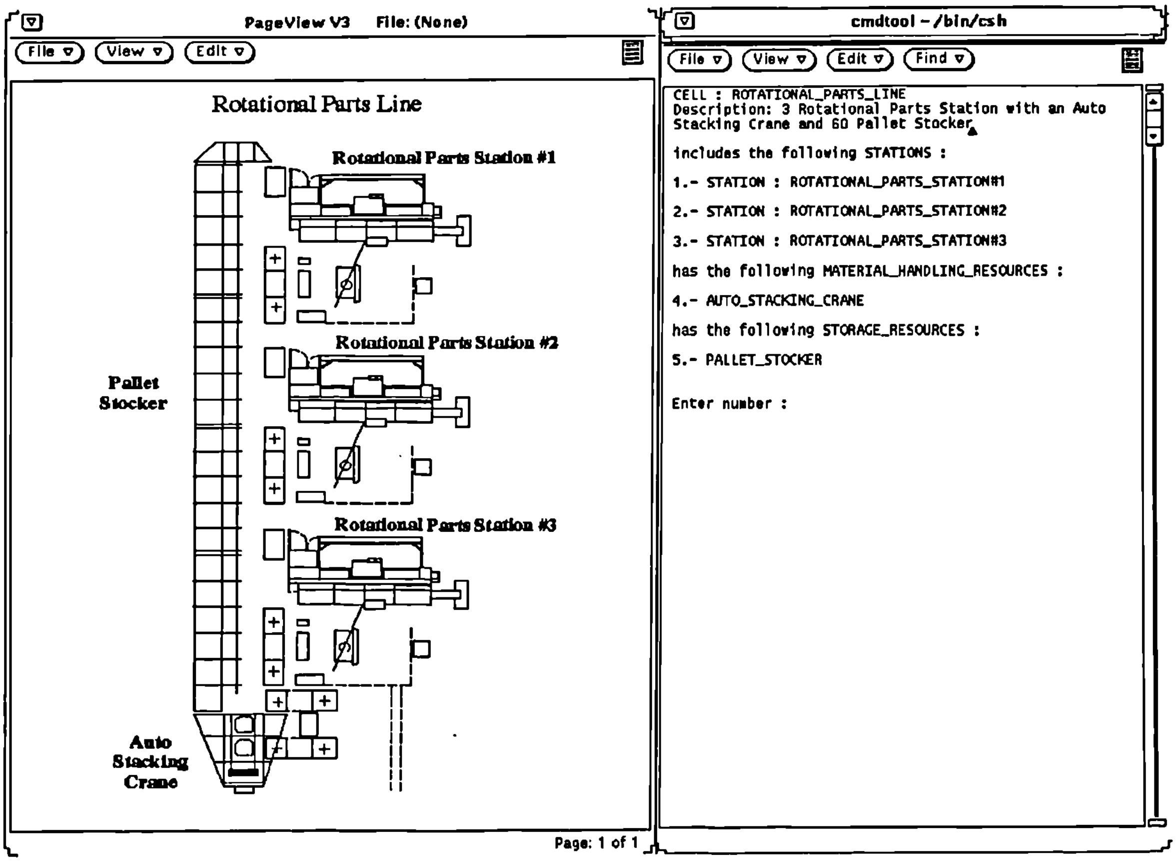


Figure J.10 Rotational Parts Line Configuration

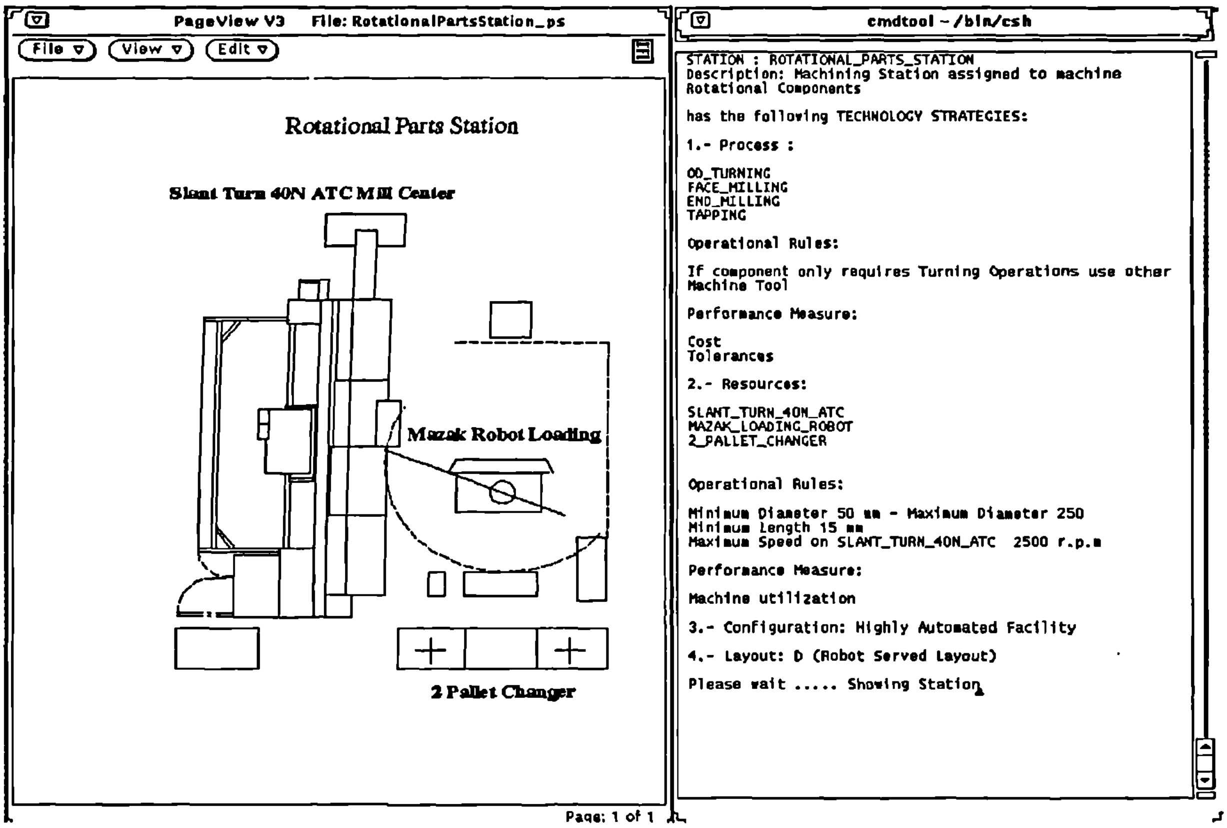


Figure J.11 Technology Strategies Description for the Rotational Parts Station

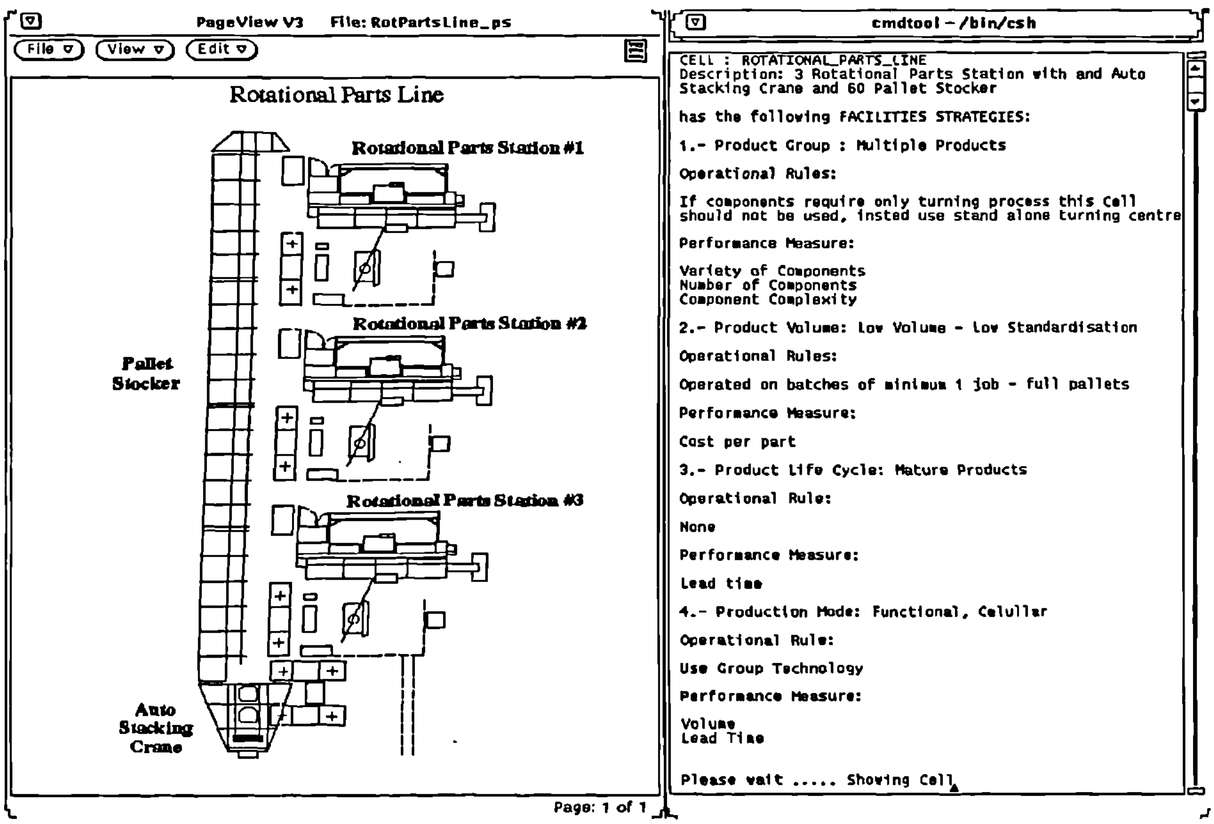


Figure J.12 Facilities Strategies Description for the Rotational Parts Line

Appendix K

Future Usability of the Manufacturing Model

K.1. Introduction

This Appendix highlights the future scenarios where the Manufacturing Model will be used. A prototype MOSES system will be developed that models the information and procedures necessary to support that part of the simultaneous engineering process concerned with design for manufacture, in particular design for machining and design for injection moulding. This application environment was selected because the content and structure of the data models will be rigourously exercised, the functionality of the moderator can be tested and the hierarchy of experts within application environments can be investigated.

K.2. Supporting Design for Manufacture

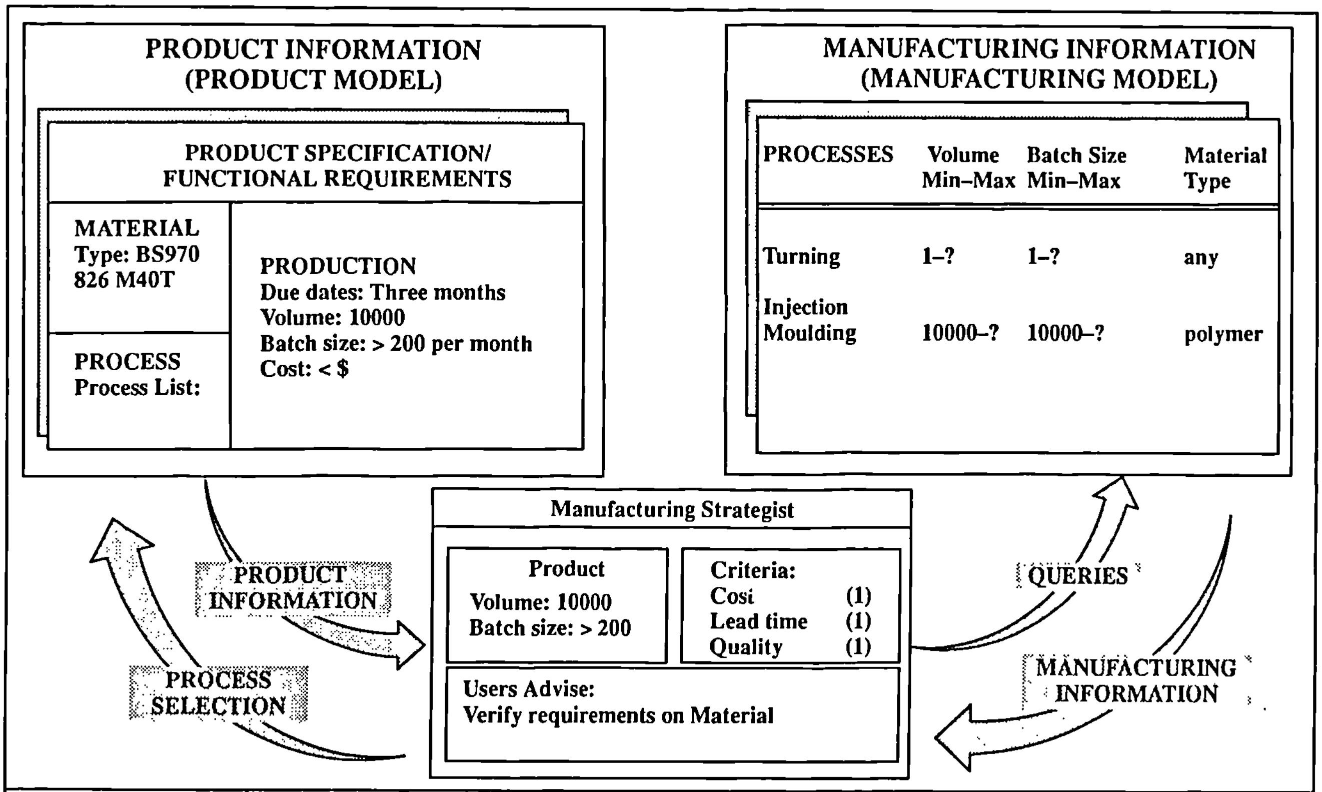
A design for manufacture application must (among other things) portray and analyse manufacturing processes and resources in relation to any given product. This requires a detailed understanding of the functional and logical interdependencies of all elements of the product and process design and is achieved in the MOSES system by the Manufacturing Strategist. The Strategist examines the content of the Product Model and, based on that information, it formulates queries for the Manufacturing Model. The response from the Manufacturing Model is analysed with the assistance of the user to enable the Strategist to make relationships between the objects and attributes in the Product Model and the processes and resources in the Manufacturing Model. The relationships can be conditional upon certain changes being made to the product model. In this way the designer receives feedback on the manufacturability of the product as it evolves and is able to influence the selection of manufacturing methods.

Figure K.1 illustrates a typical pattern of communication between the various system elements. It is assumed that the user has triggered the strategist. The strategist interrogates the product model to determine what additions and amendments have been made to product data. To use an example being tested in the prototype implementation of the MOSES system, the specification for the product may have been populated. The strategist gathers the commercial aspects of the specification i.e. due dates, production volume, batch sizes and uses this to build a query for the Manufacturing Model. The Manufacturing Model responds by presenting a list of potential manufacturing processes. The strategist also draws information on high level manufacturing strategies from the manufacturing model. The process list is ordered based on an optimisation against the criteria detailed in the high level strategies. Process capability is assessed against three criteria; cost, quality and lead–time. The processes may be near the limit of their capability e.g. barely capable of achieving the required delivery date due to tooling production lead time, near to the maximum envelope size or not ideal for the material selected. If this is the case the user will be advised of the implications and given the option to ignore the process limitations or change the product. Once satisfied with the order of the proposed processes the strategist will populate the manufacturing information element of the product model. Each process may refer to a list of resources and each resource to a list of manufacturing activities. Accountability is maintained because each process or resource in the list references a particular element of the product description.

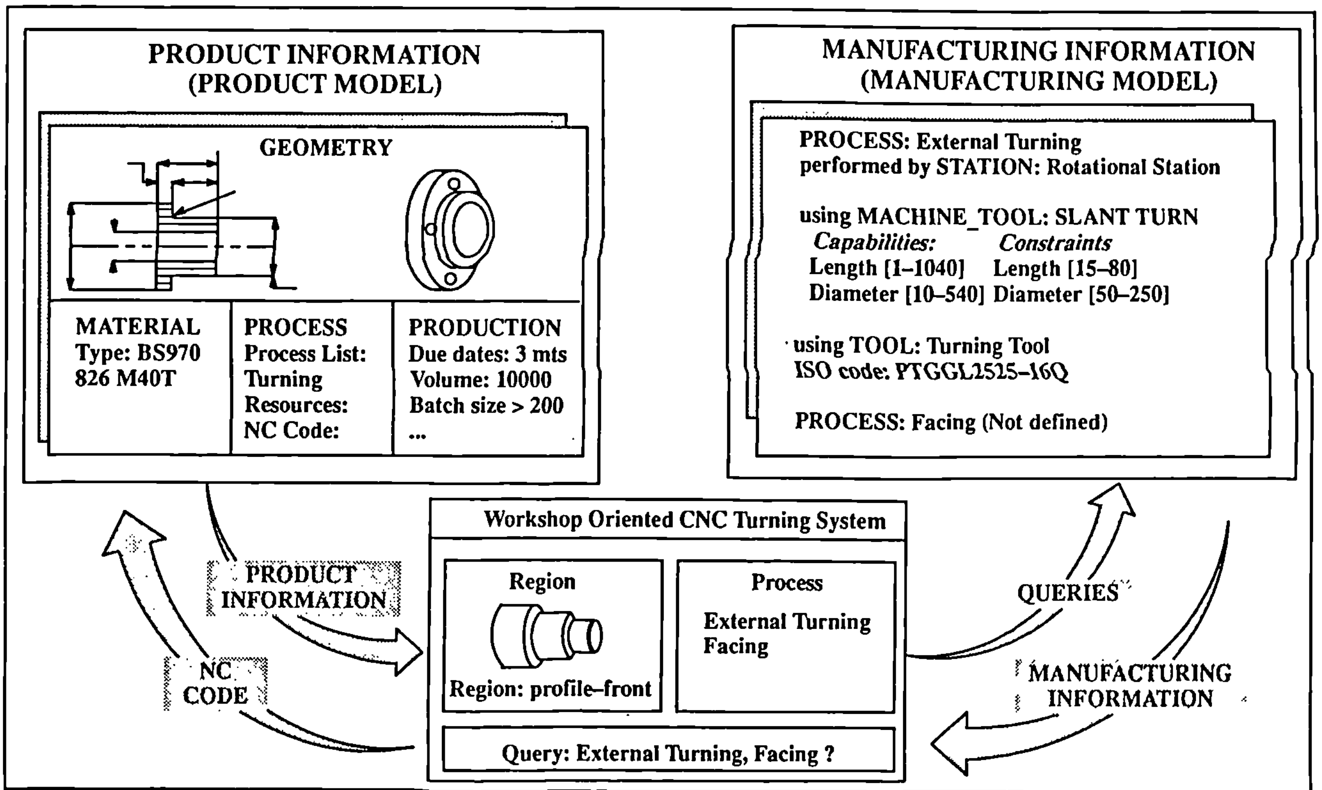
K.3. Supporting Post–Design Manufacturing Information Generation

The Manufacturing Model is ultimately intended to be used by a range of design and manufacturing applications. The information has been structured into process, resource and strategy subsets to facilitate this. The manufacturing information used by many of the application environments is similar but the interpretation placed on that information is what differentiates the functionality of the applications. For example, a NC planning system needs to know the processes that can be undertaken by a given resource as does the design for manufacture application. It is the level of detail that differs and not the basic information classification.

A Feature Based Workshop Oriented NC Planning System for Asymmetric Rotational Parts has been defined to be part of the Manufacturing Information Generation Environment. This system is able to generate NC Code based on a feature based approach (Tavakoli 1993). A set of features defining regions which can be machining are related to a set of tools and operations suitable to generate the required feature. The system can therefore select the appropriate cutting tools, machining parameters, operations sequence and generate NC Code (EIA/ISO Format) to machine asymmetric rotational components. The NC planning system is being modified to demonstrate how this type of system could be integrated into MOSES and take advantage of the information captured and represented in the Product and Manufacturing Models. Instead of establishing the relations between the regions with the tools, the regions have been related only to the process which can be used to produce the specific region. Figure K.2 shows that a relationship exists between the region profile–front and two processes: External Turning and Facing. This means that the region profile–front can be machined using either External Turning or the Facing process. These relationships enable the NC planning system to query the Manufacturing Model regarding the processes available in the manufacturing facility. For example in figure 13, the results of the query regarding External Turning and Facing is that the External Turning process can be performed by the rotational station using the machine tool Slant Turn 40N ATC 200 and tool turning tool of type PTGGL2525–16Q. However, the Facing process has not been defined. The NC planning system continues its interactions with the Manufacturing Model in order to obtain information regarding suitable tools, work–holding devises and machine tools, which must be selected to complete the required NC code.



A. Molina | Figure K.1 Manufacturing Strategist interactions with Product and Manufacturing Model | LUT–SE Research Group



A. Molina | Figure K.2 NC Code generation using Product and Manufacturing Models | LUT–SE Research Group