Evaluating the Impact of Adopting a Component-Based Approach within the Automotive Domain

By
Min Huey Ong

Doctoral Thesis
submitted in partial fulfilment of the requirements
for the award of

Doctor of Philosophy
of Loughborough University

October 2004

Abstract

Component-based technology applied to the control system of production machinery is one of the new research developments in the automotive sector. Although it is important to evaluate the technical aspects of this new paradigm, an appreciation of the impact from the business and human aspects is equally important to the stakeholders in the industry. However, the current evaluation approaches do not offer a method to capture and analyse the Component-based technology that is simple to use and produces results that are readily understood by the stakeholders involved in the process.

This study is based upon a research project at Loughborough University to look into the effect of the implementation of a Component-based control system for production machinery in the automotive sector (referred to as the Component-based Approach,) and is focused on the business and the human aspects of the approach. Within the study a structured approach to evaluate the implementation of the Component-based approach has been adopted and developed. The approach supports the various phases of the evaluation, from planning, data collection to the analysis of the data. An evaluation strategy has been formulated, comprising four stages: i) planning the evaluation, ii) evaluation design, iii) data collection methods and iv) analysis and reporting of the data. The methods used in the evaluation include: i) knowledge elicitation through interviews, analysis of documents and surveys, ii) the investigation of future implementation scenarios using scenario testing and iii) data representation and analysis using enterprise modelling approaches. The evaluation was conducted with respect to the various stakeholders in the industry, namely, the vehicle manufacturer, the component suppliers and the machine builders. Analysis was made from four viewpoints, namely: i) product, ii) process, iii) business, and iv) people.

The evaluation approach was tested on three industrially focused case studies. This has enabled the identification of the benefits and issues associated with implementation of Component-based Approach in the automotive sector.

The main contributions made by this research are:

i) The identification and development of a mixed method approach for evaluating the Component-based Approach (and consequently any technological paradigm shifts within industry), incorporating knowledge elicitation, testing and analysis.
ii) The application and evaluation of enterprise modelling techniques as an approach for data representation and analysis within technological change and assessment.
iii) The identification of the benefits that a Component-based control system will bring to the different stakeholders in the automotive industry.
iv) The identification and analysis of the business and human related issues that are likely to impact the various stakeholders in the automotive industry with the implementation of a Component-based control system in production machinery.

Keywords: Evaluation, Component-based, Automotive, Enterprise modelling, CIMOSA, Process representation, Scenario Testing.

Acknowledgement

I would like to express my sincere gratitude to my supervisor Dr. Andrew A. West for his guidance and continuous encouragement throughout the course of the research; I would also like to thank my second supervisor Dr Robert Harrison for his advices.

I am grateful to Dr Radmehr Monfared for his kind advices and credit should be paid to his work on the CIMOSA modelling diagrams which are used in this research.

I would like to express my heartfelt thanks to Daniel Vera who has always been so kind and helpful in providing invaluable guidance in the use of modelling software; Professor Richard Weston for his kind advices; I would also like to thank my other colleagues in MSI Research Institute who have helped me in many other ways.

I would like to acknowledge the Engineering and Physical Sciences Research Council (EPSRC) and Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, for providing the financial support and means for carrying out this research work.

Many thanks to the industrial collaborators, Lamb Technicon UK and Johann A. Krause Germany, for their valuable time and kind assistance during the course of this research.

I am thankful to my parents for providing me with a good education, my sisters for their constant support. Last but not least, I am most profoundly thankful to my husband, Szer Ming, for his support, guidance, encouragement and love; indeed without him this thesis would not have been possible.

Table of Content

List of Figures

List of Tables

List of Acronyms

BP	Business Process
CB	Component Based
CBA	Component Based Approach
CBSE	Component Based Software Engineering
CIMOSA	Computer Integrated Manufacturing – Open System Architecture
CAD	Computer Aided Design
COTS	Commercially off the Shelf
DP	Domain Process
EA	Enterprise Activity
HMI	Human Machine Interface
MMA	Monfared Modelling Approach
MSI	Manufacturing System Integration Research Institute
PDE	Process Definition Environment
PLC	Programmable Logic Control
SE	Simultaneous Engineering
VM	Vehicle manufacturer
VRML	Virtual Reality Modelling Language

Chapter 1

Introduction

1.1 Motivations

The automotive sector is a competitive, risky and high investment industry [1, 2]. It is constantly facing new challenges and frequently has to look for ways to reduce cost, increase production and gain a competitive edge. Enterprises in the automotive industry are actively looking for (i) new methods of production to manufacture faster, cheaper and with increased quality and (ii) ways which enable them to react quickly to market changes.

The design and the build of manufacturing machines for producing the car engines is one of the key areas in the lifecycle of automotive manufacturing. Over the years, different types of machinery had been used to meet the requirements of the particular age; for example, the Ford Motor Company developed special purpose machines for mass production of car parts, General Motors developed the transfer machine system and Japanese vehicle manufacturers developed flexible transfer lines in order to increase the variety of car models produced using the same systems [3]. The aim of the vehicle manufacturers is to develop new systems that will not compromise productivity for flexibility [4] and at the same time, reconfigure and reuse the existing systems so as to reduce cost [5]. A project initiated by the Manufacturing System Integration (MSI) Research Institute in Loughborough University has looked into adopting a Component-Based Approach (CBA) for the design and implementation of engine production machinery. The aim was to convince the automotive manufacturers and suppliers not only would the CBA meet the manufacturing requirements of the vehicle manufacturers (i.e. better, cheaper, faster), it would also be able to solve some of the reconfiguration and reuse problems faced by the industry.

The introduction of the CBA is a paradigm shift in the automotive industry. Such a change requires detailed research into its impact, such as technical impact, business impact and social impact (i.e. impact to the people in the industry) in order to provide justification for its implementation and adoption on future production systems. It follows that industrial evaluation is necessary for the implementation of CBA and this calls for the development of suitable methods for the evaluation process.

1.2 Current evaluation methods

Any evaluation method needs to support readily a study of the effect of change [6, 7]. This could possibly be done through surveys using questionnaires or interviews with the stakeholders to assess the change. However, the main problem with the classical evaluation approaches was that there were no "Component-based products and tools" currently available for the stakeholders to appreciate their use and assess the implications of adopting the approach. In addition, the paradigm has been shown to require a radical change to working practices and system design and implementation [8]. That could not have been appreciated by automotive stakeholders prior to its implementation. The current data collection methods are good and workable when evaluating a programme or project that has already been carried out and they lack the capability when it comes to predicting or forecasting the effect of change.

Review of the current evaluation methods also shown that there is a lack of simple, standard method to describe formally processes and enterprise interactions (see section 3.3). Simple flow charts, graphical representation or standard textual documents are insufficient in representing and showing the interaction of processes and resources effectively. They do not enable an easy comparison and thus an analysis of the processes and resources involved in a multi-collaborators project such as the designing and building of machines.

To assess the impact of Component-based (CB) changes on the production system design and build processes, there was a need for some form of experiments or tests to be conducted. From these experiments or testing, observations were made, looking into the changes in practices and work processes that would result from the adoption of the CBA. A set of requirements for the evaluation in this research has therefore been established. These are:

i) To be able to understand and visualise the enterprise processes of the stakeholders, the actors and the activities within the enterprise
ii) To capture fully the activities of the stakeholders and their interactions

iii) To be able to perform "what if" analyses

iv) To have an understanding and illustration of how potential users could work within the system

1.3 The Research

The problem that has been defined for the research in this thesis is summarised schematically in Figure 1.1. The research outlined in this thesis is based upon three projects undertaken by the MSI Research Institute to develop the CBA for engine production machinery in the automotive sector. The research focused on i) defining a method for evaluating a new technology i.e. the implementation of CBA for engine production machinery and ii) evaluating the business and human[1] issues related to the implementing of the CBA and the effect that its adoption would have on the automotive stakeholders.

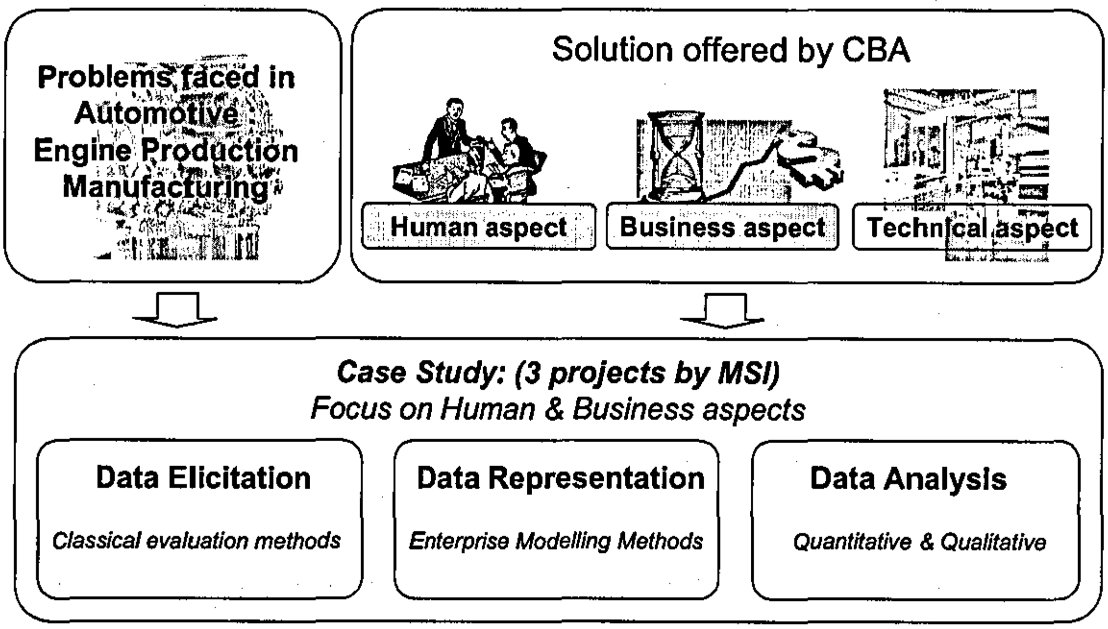


Figure 1.1 Research definition for this thesis

To evaluate the CBA, a strategy was developed which provides an overall guidance to the evaluation process adopted. This strategy consists of i) planning the evaluation, ii) selecting an evaluation approach, iii) determining the methods for collecting data and iv) data analysis and presentation. An evaluation approach consisting of three main

[1] Human issues refer to impact to people in the industry, as opposed to the ergonomic factors.

aspects was formulated, namely i) data elicitation, ii) data representation, and iii) data analysis. Mixed methods have been used to understand the domains of interest, assess future implementation and data representation and analysis (see Chapter 5). Enterprise modelling approaches have been used in the research to: i) represent the data and ii) provide a basic method for comparison of the current process and the future CBA process. Static enterprise models of the stakeholders' business processes have been created and subsequently used to develop dynamic models of the processes. In terms of data analysis, both quantitative and qualitative analysis will be conducted. Quantitative analysis will be conducted using dynamic modelling. Dynamic modelling provides the means for performing "what-if" analysis by enabling the behaviour of the system with respect to, for example, time and resources changes, to be studied. To ensure data consistency between the static model and dynamic model, a modular structure for organising processes, information and resources has been developed. Modelling method has also been adopted to represent the processes involved in developing a machine based on the CBA. Such a representation enables the potential users to appreciate the activities and the resources involved in producing CBA machines.

The design and build of complex engine production machinery currently requires multi-collaborator input. Central to the project is the vehicle manufacturer, who designs and produces the engine and subsequently assembles the complete vehicle. The other main partners are the machine builders and the component suppliers, also known as Tier-1 and Tier-2 suppliers to the vehicle manufacturer respectively. The machine builder is responsible for the design and manufacture of the production machinery and the component suppliers provide the required components for the machines.

For the qualitative analysis, all of the three main stakeholders in the automotive machine design implementation process must be included in the evaluation exercise and their individual views and experiences taken into consideration. Moreover, there are many different aspects of the project that need to be considered. These aspects include for example business / workflow processes, product development, human resources and business considerations. CBA technology is expected to have the largest impact on the machine builders since in reality these enterprises are solely responsible for the conceptual design, detailed design, build, test and maintenance of the automotive production systems. Thus the evaluation in this research has been focussed around the machine builder (i.e. transfer machining and assembly) processes with the vehicle manufacturers' and component suppliers' inputs included when and where relevant.

1.4 Organisation of Thesis

A general review of relevant research literature is presented in Chapter 2. The research focus and design of the work that forms the basis of this thesis are presented in Chapter 3 and the formulation of the evaluation strategy explained in Chapter 4. The data collection methods that have been used in this research are described in Chapter 5 and an explanation of how the enterprise modelling approach was utilised for data representation and analysis given in Chapter 6. The case studies are described in Chapter 7 and the results and analysis are presented in Chapter 8. Conclusions, discussions of contribution to knowledge and further work are detailed in Chapter 9.

Chapter 2

Literature Review

2.1 Introduction

The general backgrounds to the research topics which are relevant to this thesis are reviewed in this chapter. These are: i) the current issues associated with the automotive sector, ii) component based systems, iii) evaluation techniques and iv) enterprise modelling.

In Section 2.2, discussion about the automotive sector provides a global picture of the development in this domain with the subsequent focus being on the process of machine design and build. The problems and issues of concern that are faced will be explained. The opportunities for development of the component-based approach (CBA) in the automotive sector will be developed in Section 2.3 by referencing CBA advantages, drivers and impact as appreciated by the software industry. The CBA in the software industry is the most mature with respect to the application of the architecture and technology and hence is detailed in the discussion.

Evaluation is one of the major aspects of this work, thus a general review of the techniques for evaluation is presented in Section 2.4. Furthermore, since enterprise modelling has been used in this work as part of the evaluation method, a general background on the constructs and development process is provided in Section 2.5.

2.2 Requirement Analysis of Automotive Sector

Automotive production is the largest and one of the most complex manufacturing sectors in the world [2, 9]. The manufacturing of vehicles involves a range of industries, ranging from raw materials generation, vehicle components supply, production machinery design and development, to assembly of vehicles [2]. It is an expensive, time-consuming and risky investment [10] but contributes to employment in various industries and is an essential provider of revenue for many countries throughout the world [11].

Nonetheless, over the past decade the global automotive industry has been suffering from over capacity. Ironically, this has not inhibited the growth in production around the world. According to the data from Automotive News Data Centre [12], the global production of cars in 2000 had increased across most regions (see Table 2.1). Simultaneously, the industry is experiencing reorganisation and consolidation through mergers and acquisitions. One of the decisive factors for mergers and acquisitions is synergies, which seek to broaden the product range, increase bargaining power, generate economies of scales and cut cost through joint production and development [13-15]. In addition to global mergers and co-operation on projects is the trend towards reducing the number of suppliers and the outsourcing non-core work to sub-contractors/suppliers [15, 16]. The vehicle manufacturers (VMs) hope that by adopting this policy they will be able to have cost benefits in terms of purchasing efficiency and supply base management [17]. On the other hand, the automotive manufacturers have begun to concentrate their effort on their core-competencies of designing and assembling cars. They are now relying on the suppliers to supply "systems" which are defined in terms of function such as the steering system, the braking system and driver information systems [18]. The suppliers, in order to increase their capability, are also undergoing mergers and acquisitions in response to changes in the VMs [19].

Global Production of Cars in Yr 1999 and Yr 2000			
	2000	**1999**	**Percent change**
Africa	305,713	300,015	1.9
Asia Pacific	13,332,634	12,616,289	5.7
Central and South America	1,698, 844	1,418,787	19.7
Eastern Europe	2,646,446	2,470,134	7.1
Middle East	571,430	464,041	23.1
North America	8,372,108	8,261,117	1.3
Western Europe	15,319,360	15,373,340	-0.4

Table 2.1 Global production of cars in Year 1999 and 2000 (adapted from Automotive News Data Centre [12])

2.2.1 Challenges in the sector

The automotive industry is constantly subjected to and in the process of accommodating changes. These changes are generally attributed to the following factors: i) globalisation, ii) reduction in the product time to market, iii) the need for product differentiation and iv) cost reduction pressures [1, 20].

Globalisation

Globalisation developed rapidly in the 1990s [15]. Many VMs faced slow growing and saturated markets in their home countries. At the same time, they faced competition from VMs from other countries (for example Japanese cars entering the US market) [21]. In addition, there was excess production capacity. These factors prompted the VMs to shift their attention to the emerging markets such as Latin America, North and South Asia, China and India, where the markets are forecasted to grow at more than double the average industry rate over the next ten years [10]. Furthermore, the production costs (particularly wage costs) are often lower in the emerging markets and in addition some countries offer trade benefits (such as North America Free Trade Agreement, i.e. NAFTA and the European Union) which are also incentives to the VMs. At the same time, the suppliers to these VMs follow the footsteps of the VMs, so that they can respond to the needs of the clients wherever they are located [10, 16, 21]. The internationalisation trend has led to the global sourcing of suppliers by the VMs. VM's now source from worldwide suppliers who are capable of offering them optimal cost, quality and service terms [16]. These suppliers could either be located in the same country or located in other countries.

Reduce Product Time to Market

Sterner competition has also forced the VMs to introduce more models of cars onto the market with an ever shortening product life. It has been found that the delaying of launching a product on the market is costly to the VMs [22]. For a product which has a life of five years, a six months delay in its launch will cause a 32% reduction in profits [22]. Already some VMs are trying to shorten the duration of the development project from 48 months to 24 months and indeed some are targeting 21 months [23]. Already two Japanese VMs, Toyota and Honda, have refined their product development cycles to introduce new models in as little as 24 months as compared to the three to four years previously [24]. The shortening of product life means that there is less time available to recover the research and development costs. Losing out in bringing the products to market in competitive timescales, however, would mean a loss in revenue to competitors.

Product Differentiation

Nowadays, many vehicles on the market have similar functions and performance characteristics [10]. To increase market share, the VMs strive to offer differentiated products that will win over the customers. People carriers, sports utility vehicles and off-road vehicles are some of the results of product differentiation [25]. VMs strive to offer vehicles that are better adapted to new consumer attitudes such as combination of comfort and practicality of minivans in Europe and light trucks in the United States [16]. Through the acquisition and merging with other automotive companies, automotive manufacturers hope to be able, not only achieve economies of scale, but also produce a car for every segment of the market or every type of consumers [14].

Cost Pressure

Automotive production is highly capital intensive and risky, requiring huge investments in product development, model-specific tooling and production facilities. For example the recent I4/ I5 diesel engine project within Ford was costed at $2.5 billion (private communication). The increased varieties of car models and the increasing complexity of the car increases the cost of product development[14]. In fact, cost reduction and increased product variety are the two main objectives the VMs have identified as key to strengthening their competitiveness [16]. To reduce the product development cost, VMs have adopted the strategy of having "common platform" [15, 26, 27]. The idea is to produce and market a "common" vehicle for the major markets by sharing a large number of components. Adjustments to aesthetic components enable response to local adaptations in the markets to be achieved. This common platform policy enables the VMs to enjoy economies of scale, through: i) the joint production of engines and new models across the different markets and ii) eliminating redundancies in development costs and the number of platforms offered in market [24, 28]. Such a strategy is implemented not only within the organisation itself, but also among other VMs within the same consortium, for example, the compacted graphite iron engine produced by Ford, is also supplied to Land Rover, Jaguar and Peugeot-Citroën Group [29].

2.2.2 Lifecycle of Machine Design and Build

Against the backdrop of globalisation, competition and increasing cost pressures, the process of designing and building the production machinery for the production of car engines, (i.e. the main focus of this research work), has no doubt been subjected to similar pressures.

In this process the stakeholders involved are: the end user, the machine builder and the component supplier. End users are the automotive production companies that design, build and distribute new automotive vehicles, i.e. vehicle manufacturers. Machine builders, also known as original equipment manufacturers (OEMs), are known as the Tier-1 suppliers to the end users; they design, develop and install the production machineries for the end users. The component suppliers are producers or vendors of control hardware, such as servo drives, hydraulic and pneumatic actuators and sensors. They supply the hardware to the machine builders and are known as the Tier-2 suppliers to the end users [30].

A schematic of the engine product design process is as shown in Figure 2.1. Currently, the product development period typically takes 42 months (i.e. almost twice as long as the targeted 21 months). The process begins with the conceptualisation of a new car model and the engine associated with it. This leads to a period of product design for production and even before the concept is finalised, the end user will contact the machine builder to begin the process for design and building the engine manufacturing systems. It is noted that the overlapping of the processes is necessary to compress the timescales but is, however, susceptible to time and cost risks associated with late changes to the product and process. It is vital that these issues can be addressed. The production machinery design/build process is deemed as the riskiest since it is hard to change the software and the hardware in current systems, especially late within the development lifecycle. It has been reported that late changes are the major source of cost variance in current projects [31].

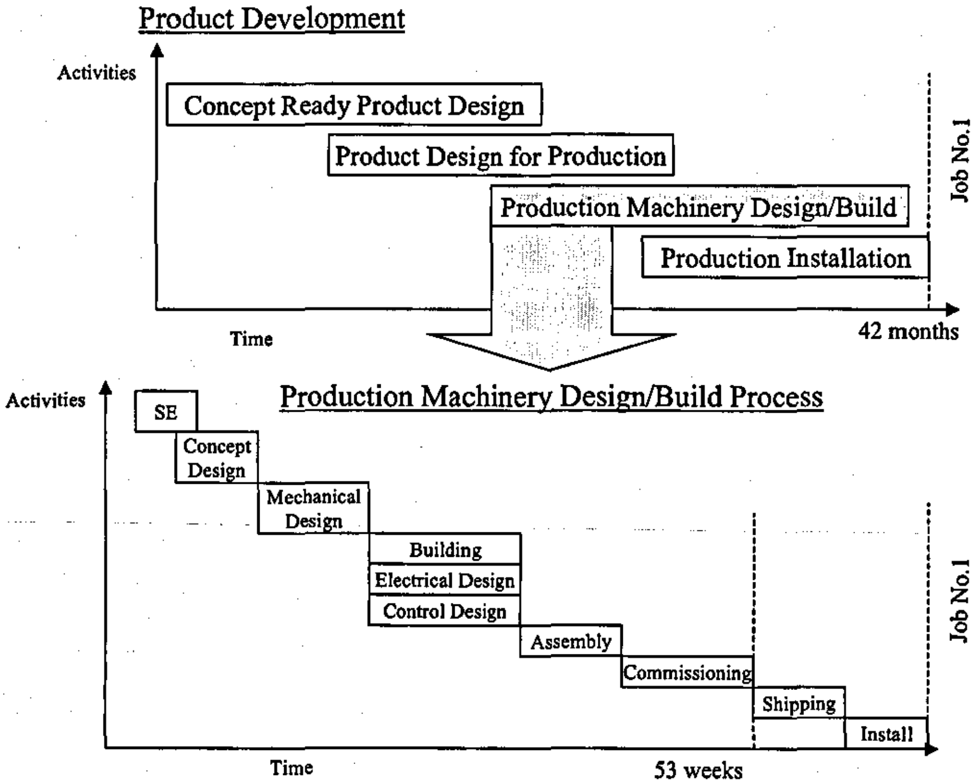


Figure 2.1 Product Development and machine design and build process.

The process of Production Machinery Design/Build (detailed expanded in the lower diagram in Figure 2.1) typically takes 53 weeks to implement. At the start of the project, the VMs will invite the machine builders and component suppliers to participate in simultaneous engineering meetings to identify technical and economic issues associated with the new engine project. This is followed by the conceptual design of the machine production system. The mechanical design, electrical design, control design, building, testing and installation of the machines are all carried out by the machine builder at their sites. The end user sends its engineers to inspect and monitor the processes and system build at specific stages with the development lifecycle. At each stage of the development, approval from the end user is required in order to proceed to the next stage of the development. The machine builder's main task is to design and implement the mechanical, control, electrical and hydraulic systems. Some machine builders also sub-contract certain parts of the work to specialist component builders or system designers.

When the machine has been assembled, tests are conducted to ensure that the machine meets the production requirements of the end user, e.g. quality, throughput, error recovery, functionality, user interface capability and diagnostics and maintenance support [32]. When the final machine has been approved by the end users, the machine is stripped down, shipped and installed on the shopfloor of the end user. Machine builders also send their engineers to undertake on site commissioning and testing of the system. Following the completion of the production machinery realization process, the end user is ready to produce the first engine (known as Job 1). The system will still be under constant inspection by the machine builders for several months to ensure that the machines are in good working order. It is not unknown for machine builders to be involved in the operation and maintenance of the production system throughout its complete lifetime.

2.2.3 Problems and Issues involved

Extensive interviews with the stakeholders have identified several issues and problems associated with the current process of machine design and build. These have been summarised graphically in Figure 2.2.

i) Paper based Documentation requiring Translation

The documentation within the machine builder enterprise is largely paper-based. Different engineers will produce different sets of drawings and documents and these are passed on to other engineers further down the process chain to develop as required. At every stage, interpretation and translation of the documents are undertaken to suit individual requirements. This process of interpretation and translation is inevitably vulnerable to errors. For example, within the same organisation, different departments have different naming conventions for the devices or equipment. Consequently, meetings have to be held to sort out the discrepancies and standardise on the name for the equipment. In many cases, errors are not discovered until late in the process during hardware implementation.

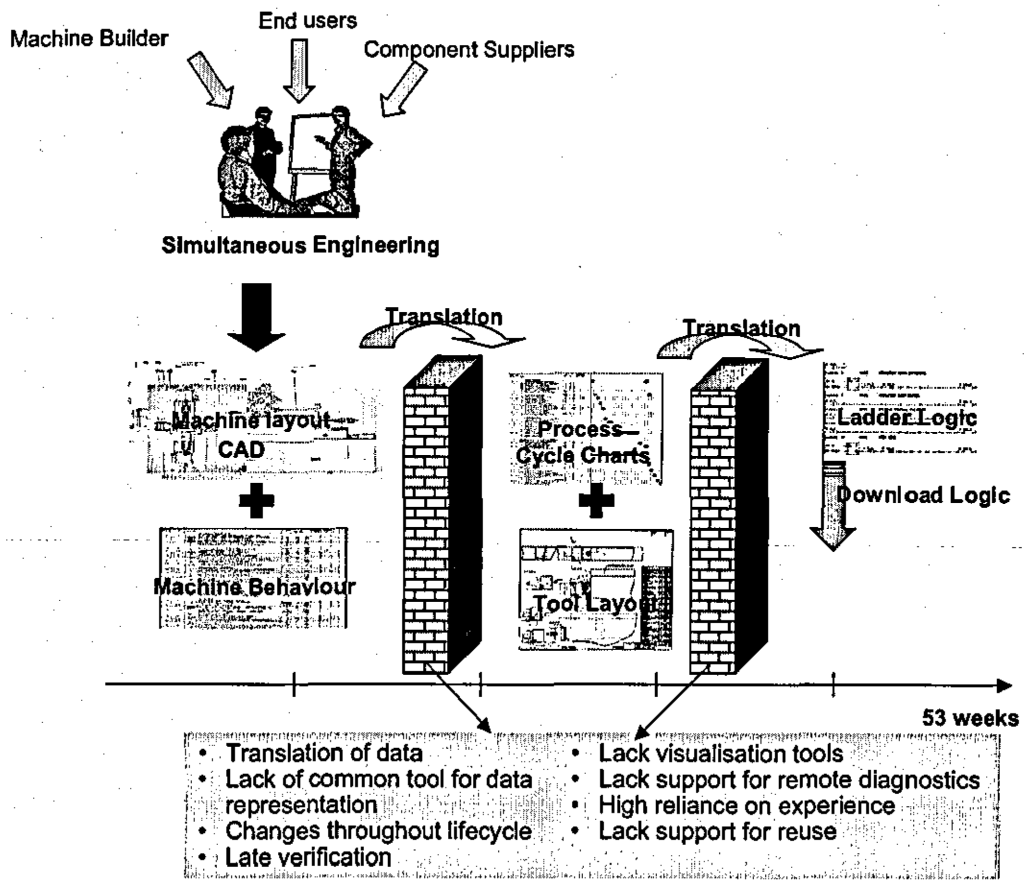


Figure 2.2 Problems faced during the process of machine build and design

ii) Lack of Common Tool for Data Representation

The paper based documentation issue described in the earlier paragraph also illustrate a lack of a common data representation in the process. Engineers involved in the process perform their roles by using a set of specialised tools and methods that include computer aided design (CAD) tools, vendor specified ladder logic programming tools and charting of timing diagrams [33]. For example, the process description of the functionality of the production machinery is given to the mechanical department in the format of a set of words that indicate the movement of actuators and the states of sensors. This is re-interpreted into timing diagrams (i.e. diagrammatic representation of the time sequence of sensor and actuator states similar to Gantt charts for project activity progression [32]) by the mechanical engineer who will then pass it on to the control engineer. Based on the timing diagram, the control engineer will generate the equivalent control timing diagrams and re-express them into ladder logic programming for the programmable logic control software [33]. This is illustrated schematically in Figure 2.3. It is evident from Figure 2.3 that the specification of machine behaviour cannot be shared easily among all the different engineers (i.e. process, mechanical and control engineers). Although there is a common repository for certain information such as scheduling, change management and purchasing, each department maintains its own data sets pertaining to their specific requirements. There is far too much repetition of effort and re-interpretation within the current process.

It could be seen that the tools used by each engineering department could not interoperate effectively without significant development effort. Integration of the large range of current systems would be extremely time consuming and costly, and if changes were required, the process of interpretation and re-interpretation would still to be repeated, and as mentioned earlier, the translation and interpretation process would still be vulnerable to errors.

Globalisation has resulted in the fact that the VMs activities (such as design, production, marketing, distribution) are geographically distributed around the world. For example in Ford's I4/I5 engine production programme, the "engine product engineering" was conducted in Japan, the "business office" was set up in Germany, "manufacturing engineering" was undertaken in UK and five separate manufacturing plants were implemented worldwide [34]. Such a distribution of activities requires good communication and coordination. Furthermore there is an implicit need to have a common data representation that could coordinate and support the inter-working and decision making of the distributed engineering teams [34].

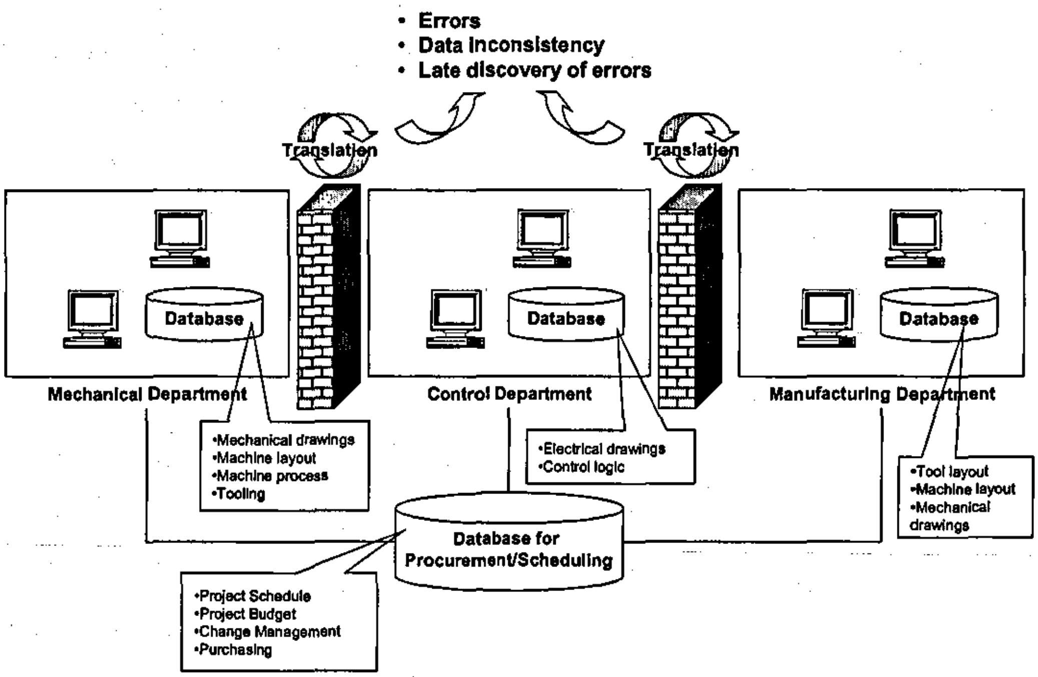


Figure 2.3 Graphical illustration of the problems in translation and the lack of common data representation

iii) Changes throughout the lifecycle

Although the SE process conducted at the beginning of the production machinery design/build process is aimed at the early identification of functional, technical and financial issues and possibly developing a solution for them, changes are still inevitable throughout the process. Changes that are necessary at the later stages of the project are most costly because all earlier work needs to be re-evaluated and new investments of resources are required. Since increasing market competition has forced the end user to introduce immature products onto the market, even though SE had been conducted, the production design cannot be finalised by the machine builder almost until shipment to the end user production facility. Hence, changes occur frequently during the development of the production system [35]. Figure 2.4 shows the tracking of different type of changes that have occurred during a particular machine design and build project, in which the product design was immature (names and date has been omitted for confidentiality reasons). There are a total of eleven types of changes and only six of the most costly changes are shown in the graph. For example, a product change (i.e. changes to the design of the engine block) would imply re-assessment of the machine design which will include changes in mechanical design, electrical design, control design and project scheduling (due to the re-purchasing and delivery of equipment). The changes resulted in the postponement of the delivery date, and the cost implications associated with such changes were high. The changes not only affected the machine builder, but had an impact to the end user as well, both in terms of late delivery to the market of the product and associated loss of revenue. Nevertheless, such situations are expected to continue due to the increasing global market competition.

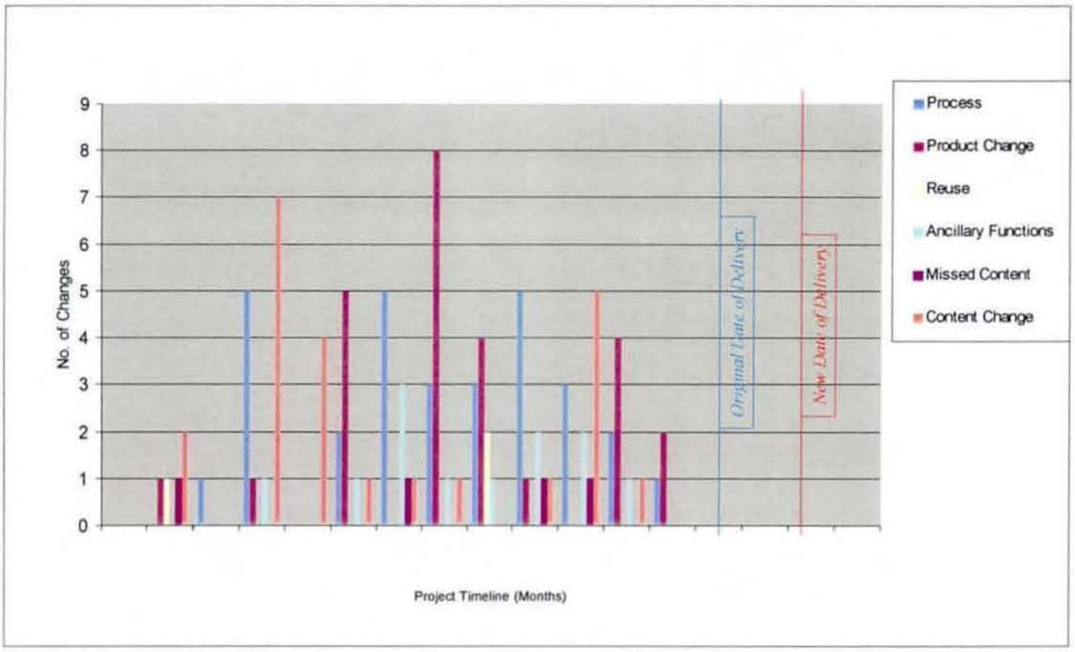


Figure 2.4 Amount of changes that occur during the design and building of production machinery.

iv) Late verification of automation system

Currently the verification of the automation system is only possible when the software and hardware systems are assembled. Throughout the design process, there are no tools available for the engineers to verify their design prior to build. The physical machine design and the control system design activities remain isolated from one another. Verification can only be carried out during commissioning when all the mechanical, electrical and control parts have been integrated. Delays within the earlier processes often have the result that time is limited at this stage and hence, successful projects very much depend on the engineers' experience to get things right first time. Any rectification of inconsistencies can be very costly in terms of time and engineering effort required. Based on interviews with the machine builders, it has been found that parts may be missing at this stage, due to late delivery. Under such circumstances, manufacturing engineers might have to "tweak" the machine such that it can still run without the missing parts and replace the parts as and when they become available. At other times, late requests for changes have also resulted in the machine builders having no time to test the impact of the change. They often have no option but to implement the change directly when setting up the machine at the end user's site.

v) No Simulation/Visualisation Tools

Currently, it is not a common practice for machine builders in the automation sector to use simulation for the verification of machine design and process simulation even though such techniques are already available [36, 37]. Although they acknowledge the advantages of simulation, the creation of virtual models of the system requires extra resources (i.e. both time and human resources) which due to cost and time pressures they are reluctant to take on. Simulation is also seldom reliable because the behaviour of the virtual system, which is constructed separately without the associated control behaviour of the targeted system, often does not accurately mirror the true system behaviour.

vi) Lack of Support for Remote Diagnostic

Machine builders have to deploy their engineers to the end user's production site for up to a year following the delivery of a new machine. The engineers' jobs are to make sure that the machine remains in good working order and also to train the operators and transfer maintenance skills to onsite engineers. Thereafter, whenever there is any problem with the system that the end user cannot solve, they call on the machine builder which frequently results in the dispatching of their engineers to the production facility. Very often, the engineers go to the site without an accurate picture of the problem, as it is difficult for the engineers to describe precisely the problem over the telephone.

Although the machine builders are able to logon to the end users' machine remotely, they can only access the control software of the machine. At the moment, they do not use 3D model simulation of the machine in real time. Feedback from the machine builders shows that they would prefer to have a visualisation of the machine in real time and check the performance against the control software coding but again are reluctant to incur the cost that they assume would be associated with this capability.

vii) High Reliance on Experience in Design and Implementation

As is evident from the above, the process of design and developing the production machinery is highly experienced based. Developments of machines are largely similar across projects and the process relies heavily on the expertise of those who have participated in similar projects in the past. Engineers who are familiar with previous projects, reuse as much of the design as possible, including mechanical, electrical and control design. In many cases, they just "cut and paste" fragments from previous projects, making appropriate alteration to suit the needs of the new project.

This high reliance on experience is a major problem since it is not unknown for the highly experience engineers to leave the company without the knowledge being passed on to other relatively less experienced engineers. There is a requirement to capture this experience and knowledge so that it could be easily shared among the next generation of engineers.

viii) Lack of support for reuse

The lack of support for reuse is related to the previous issue of a current high reliance on experience. Interviews with the machine builders have shown that there is a lot of reuse in terms of mechanical, control and electrical design. As one of the machine builders had mentioned, about 70% of their control functions are repetitive, however, the ability to reuse these previous work is usually undertaken by "asking around"; it depends on whether anyone in the team could remember whether a similar design had been used in previous projects [38]. The designs are reused exclusively using the "cut and paste" method. This shows again that having previous experiences of working in previous projects is important (i.e. highly reliance on experience). Much as the machine builders would like to reuse the designs in a more systematic manner, there is currently no such support available.

From the hardware perspective, end users are expressing interest in upgrading and reusing their current machines for new projects so as to reduce new product development costs. However, there are several issues associated with the reuse of the old equipments. First some of the devices might be outside of their useful product life and it is not feasible to reuse them. Secondly, the cost of re-engineering the old devices might be prohibitive. Feedback from the machine builders has shown that the engineering cost of reusing older devices is high because they have to modify these devices so that they will be compatible with "upgraded" technology. They are somewhat reluctant to reuse because in most cases, the machine builders have to absorb the engineering and compatibility costs. Thirdly, maintenance cost for older machines is high. Fourthly, older devices do not have warranty, only the new devices are warranted by the component suppliers. Lastly, there is the logistical issue for re-engineering of the older machines, which the parties involved have to decide where will the re-engineering of the machine be carried out.

There has to be incentives for the machine builders to support the reuse of the older equipment, such as, for example, a reduction in development time and the ability to reuse the equipment easily without much re-engineering.

2.2.4 Manufacturing Requirements

Based on the above discussion, a set of desired attributes for automotive manufacturing systems have been identified:

- To be able to respond and react quickly and accommodate changes throughout the lifecycle.
- A high degree of reuse is desired throughout the lifecycle of machine design and build. This includes reuse of automation hardware, control software, engineering knowledge and best practices acquired from previous projects.
- A common data representation is required to support the various phases of the lifecycle. A consistent data representation would help to reduce repetitive work of interpretation and translation of design specification. This common representation seems all the more important when the partners in the projects have differences in (i) geographical location, (ii) levels of experience and understanding, and (iii) cultural and language backgrounds.

- Visualisation of system behaviour through modelling and simulation prior to installation is desired. This would enable the control engineer to validate the system before the physical assembly.
- There is also a need for an approach that will enable the machine design and the associated control behaviour to be available to all interested parties throughout the lifecycle. It has to be in a format in which the users can easily relate to.
- A more integrated support for the system diagnostics and maintenance is required.
- Business process and functional benefits of innovations, new technical architectures and approaches need to be appreciated readily by non-technical managers to ensure commitment, uptake and investment are achieved.

2.3 Component Based

It has been the aim of the MSI Research Institute at Loughborough University to demonstrate the applicability of the CBA in achieving the desired attributes within the automotive industry outlined in the bullet points in the previous section. Component based approaches to system development have gained much momentum from the software industry. It has been adopted and widely used in software development [12]. The focus of this thesis is to not to present the technical or implementation details of the CBA, but to provide the business impact of CBA's within the automotive domain. The following discussion provides a brief description of component-based software engineering and reviews the advantages that the component-based approach has brought to the software industry. It is hoped that this discussion will enable the advantages that the approach could provide to automotive manufacturing to be appreciated. In the software industry the CBA is only applicable to the *software* and does not include the *hardware* aspect required within automotive production systems. It is nevertheless a mature component based domain (in terms of the application, architecture and technology) that could be used as a reference for automotive manufacturing.

2.3.1 Component Based Software Engineering

The engineering of component based software systems, which enables the building of systems that have predictable properties in a repeatable way has been termed as Component Based Software Engineering (CBSE) [39]. The foundation of the CBSE paradigm is that in large software systems, there are certain parts which reappear regularly and these common software parts could be written once, and in many cases, reused many times. These reusable parts in a system are structured into components. CBSE has shifted the emphasis from programming software to composing software systems [14]. Furthermore, CBSE encourages the use of "commercial off the shelf" products, which are aligned with the "buy, don't build" philosophy advocated by Brooks [40].

There are many definitions of a component [12, 41-47]. The definitions vary according to the different applications domains and thus each definition has different focus [48]. In general terms, a component is a self-contained part or subsystem that can be used as a building block in the design of a larger system [12, 14, 33, 44] and provides specific services to its environment across well-specified interfaces [12]. The different definitions of components do not affect the basic characteristics and the advantages that adopting the CBA has enabled.

2.3.2 Characteristics of Components

In software development, the characteristics of a component include: encapsulation, interface, reuse and plug and play capability.

I) Encapsulation

Encapsulation has been defined as "the inclusion of one thing within another thing so that the included thing is not apparent" [49]. In the CBA, encapsulation is the process of hiding the implementation or code that is required for the component to function and this code is accessed by an interface exposed to the user of the component. Users can adhere to the interface to use the component without having to know how the component accomplishes the task. The main idea of encapsulating a component is "Don't tell me how you do it, just do it" [49].

ii) Interface

Each component is accessible to the user via its interface(s). The interface enables the user to understand in general what the component can do [45]. It is thus also a means by which the user "communicates" with the component. The fact that interfaces have been clearly defined enables a component to be reused even if the user does not know the programming language or platform it uses internally; the same specification may be implemented in several different ways[37].

iii) Reuse

One of the most important arguments in adopting CBA is software reuse. It follows that in a large software system, there are certain parts which have sufficient regularity, that these parts could be written once and reused many times. These parts could also be reused in other systems which require similar functionality. Veryard [44] has broadened the definition of component reuse to include the reuse of knowledge and intellectual property. Many of the advantages of the CBA are associated with the savings in cost and time when components are reused.

iv) Plug & Play

To "plug and play" a component means to plug the component into a device (or another component) and have the device recognise that the component is present. The user gets expected services from a given configuration of devices. In addition, plug and play implies that the user is able to substitute easily the component with another component [50].

2.3.3 Drivers for CBA in Software Industry

The CBA has been perceived as key for developing systems in a cost-and-time effective manner [12, 14, 39, 43, 45, 50]. The advantages of the CBA drive its development in the software industry and these are closely related to the characteristics of the components. For example the CBA has been claimed to provide the following advantages to implementers and end users:

i) Improves productivity

One of the arguments is that CBA improves system designers' and system implementers' productivity. This stemmed from the assumption that if components (i.e. software code) can be reused, implementers do not have to start from scratch and productivity will increase. The Software Engineering Institute at Carnegie Mellon University conducted a thorough survey and concluded that the reasons for embracing the CBA is economic rather than technological [39]. Their survey concluded that the software market has the maturity and an increasing size to support software reuse and this provides sufficient economic incentive to embrace software reuse.

ii) Improves time to market

The reduction of time to market is closely related to increased productivity. The fact that a component has well-defined boundaries implies that parallel development is possible. Multiple teams can develop different components at the same time [43, 45, 48]. Building software from scratch takes considerable time. By assembling components that are already available, capabilities can be delivered more quickly. Where coding is required, the modularity and encapsulation of the system make it easier to understand the logic and make the appropriate changes more quickly [45, 50].

The reduction in time to market can be attributed to two main reasons: well-structured application architectures and higher levels of abstraction. Since the CBA requires a structured approach towards system implementation, the components are configured and interconnected within a well-defined application architecture. This well-defined architecture helps to streamline the process of system building and systems could be built within a relatively short period of time [39]. Secondly, the use of component technology increases the level of abstraction and more complexity could be packaged into a component framework. At the same time, because the component framework distinguishes between infrastructure builder and application builder, the skills required for development are differentiated and separated.

iii) Lifecycle cost and risk mitigation

Cost savings from CBSE results not only from the ability to reuse components but also from the reduced integration effort [50]. Reusing software saves the programmer's effort from developing from scratch and thus a saving in development cost. Maintenance cost is also reduced because of the ability to isolate problems and make changes to their implementation without affecting the rest of the system. As noted by Butler et al. [50], the cost of modifying a component is 20% that of developing it from scratch. The savings arise from leveraging all of the requirements analysis, design and testing effort expended to create it. Using components that have been proven within a framework eliminates unknowns and risk associated with technical, cost and scheduling [39].

iv) Assured quality and improve reliability

Components that are reused many times are likely to be more reliable than new, hand-coded components [43]. This also implies that the quality of components is assured with reused components, because these components have been proven that they work. By the same argument, applications that have been reused many times will be more reliable and the quality assured than newly designed and implemented applications.

v) Adaptable and flexible infrastructure

The common and interoperable information infrastructure of CBSE set the foundation for an adaptive architecture that enables faster time to market of new capabilities [50]. The clear and well-defined structure of a component based architecture makes applications easier to understand and maintain. The encapsulating functionality of a component facilitates "plug and play", making it easier to replace functionality as business requirements change.

2.3.4 Enablers in Software Industry

The move to CBA's in the software domain has provided many benefits to the industry. However there are many issues that have enabled the benefits to be realised. Typical driving forces behind the success are:

i) Available of standards and available technology

There is an general acceptance of the standards set by market leaders [12] and this provide a good foundation and gives enterprises confidence to design their architecture based on components. Based on the architecture, end users can determine how to acquire the components that are required for their application areas, purchase them from the marketplace, build them internally, reuse them from legacy applications, or commission their development by outside providers [50].

ii) Commercially available components

With the wide acceptance of component concepts, there are a large number of suppliers of commercially available components. These products, also known as 'commercially off the shelf' (COTS) components, are "ready made" components, which can be used and integrated readily into a system [42]. Independently developed components based on widely adopted component technologies such as JavaBeans, EJB from Sun and COM / DCOM (Components Object Model / Distributed Components Object Model) from Microsoft, allows components to be deployed and collaborated with one another [12].

iii) Best practices

In the software industry, the component concept has been evolving since 1990s and some companies have succeeded in applying the concepts in profitable applications. Knowledge from these companies has been shared within the industry and increasingly this has been used as the basis of an industry-accepted interoperability standards [12, 50]. Along with the standards come best practices derived from successful applications experience. These practices are exploited and provide a good start for adopting the component concept.

2.3.5 Impact of CBA in the Software Industry

The implementation of CBA has represented a major paradigm shift in the software industry. It brought about several changes and several authors [12, 41, 48, 50, 51] have written about the impact it has on the industry. In general, there are three main user groups in the software industry: the component providers, component integrators and the integrated system users [41]. Brereton and Budgen [41] have developed a framework for analysing issues of the CB development in the software industry. Under this framework, there are three classifications of issues or viewpoints: product, process (i.e. development process) and people. The people viewpoint is further sub-divided into business and people. Each of these issues is viewed from the perspective of the component providers, component integrators and the integrated system users. The author has adapted the framework created by Brereton and Budgen [41] to summarise the impact of the implementation of the CBA in the software industry (Figure 2.5).

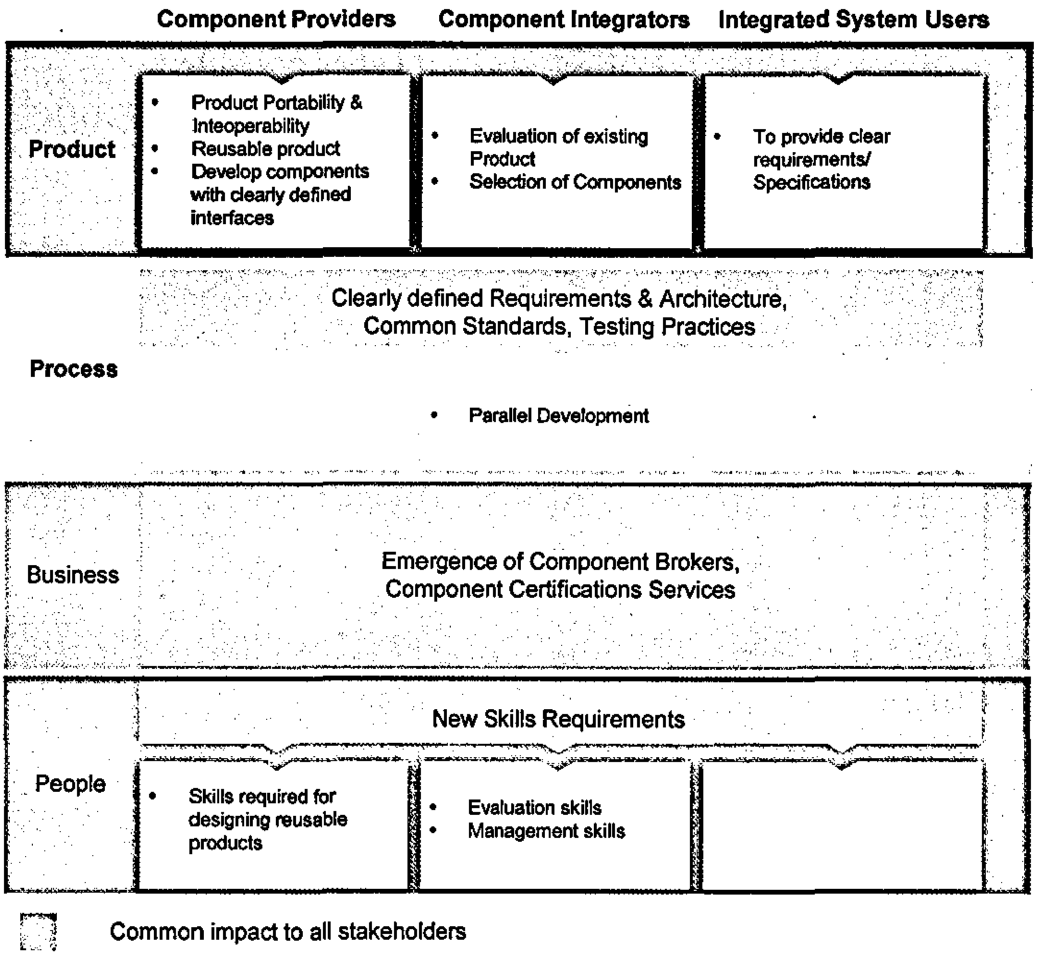


Figure 2.5 Impact of CBA in the software industry from the perspectives of the stakeholders i.e. component providers, component integrators and integrated system users. (adapted from Brereton and Budgen 2000 [41]) [2]

2.3.5.1 Product Issues

The component providers and the component integrators are the two main groups that are affected from the product perspective.

Component Providers

New Product Development Strategy

The component providers, as suppliers of components, build components for a variety of applications. They have to identify the common components that are often used in a system. These components have to be portable and interoperable across

[2] Similar diagrams have been drawn to analyse the advantages and the impact of implementing CBA in the automotive manufacturing domain in **Chapter 8.**

different systems. Increasingly, component providers are developing products based on a set of standards, component platforms and component tools which will support the CBA [39, 48].

Design Reusable Products

The component providers have to design components that are reusable. Such a move requires the component providers to develop components that conform to a common standard and provide a clear description of the components.

Component Integrators

Change of Development Focus

The use of commercially available components has shifted the focus of the component integrator in software development from 'specify, design and implement' to 'select, evaluate and integrate'. The component integrators have to consider commercially available components when designing systems. Their responsibilities lie more with evaluating the existing components rather than developing new components. They have to select and evaluate products based on technical factors (functional capabilities, behaviour and non-functional attributes or constraints) and commercial considerations (such as cost, availability and supplier reputation) [41]. Additionally they have to ensure that components can be adapted or reconfigured to achieve interoperability.

Integrated System Users

Clear and Precise Requirements Specifications

The integrated system users have to define their requirements clearly. They have to specify their requirements in a way that enables the integrators to be able to identify and undertake the trade-offs necessary in selecting suitable components [41].

2.3.5.2 Process Issues

From the process perspective, there are common impacts amongst all of the three groups. Researchers have pointed out that the development of CBSE divided the previous task of system development into component development and component integration processes [51].

Common Impact

Common Industry Standards

Industry wide, a common standard for component development has emerged. This provides a good foundation and gives the integrated system users confidence to design their architecture based on components. Standards have provided a source of stability in the industry [42], without which, any changes can impose unanticipated and unpredictable consequences.

Clearly Defined Requirements and Architecture

The implementation of the component-concept requires a pre-defined implementation architecture for components so that they can interoperate with other components or other frameworks. The interactions between the components had to be clearly defined. These requirements had to be structured and defined. During traditional developments, system integration used to be at the end of implementation phase but it has now become the centre of the development under CBA. System integration has to be planned early and continually managed throughout the development process [42].

Testing Practices

As noted by Sparling [48], the strategy for testing a component-based system is different from traditional testing methods. Theoretically, components could be tested individually, but in practice, they are tested based on applications.

Components Providers

Clearly Defined Interfaces

Components providers have to define the interfaces of the components such that they could be reused, adapted or replaced easily, i.e. plug and play functionality.

Components Integrators

Parallel Development

Due to the clearly defined structure and the encapsulation of a component, integrators are able to achieve parallel development. Since certain parts of the system could be acquired from the component providers, the component integrators are able to develop systems in parallel, significantly compressing timescales. Within the organisation, sub-groups of the developing group can develop the required components based on the specifications and deliver a complete, tested and robust component, ready for integration.

2.3.5.3 Business Issues

The business issues do not relate particularly to any one of the three groups. They are general observations of the industry rather than specific issues for individual groups.

Emergence of Component Brokers

The development of the CBA has provided new business opportunities in the market. The separation of component development and component integration provides an opportunity for the creation of a new component broker role in the market. The component brokers sell and distribute software components. They are the mediator between the component providers and component integrator [51]. Their services may include helping the integrators to find and acquire software components.

Component Certification Services

There is also the emergence of components certification services [39]. Some companies are providing certification tests for components to ensure that the components conform to certain standards, provide particular qualities of service or exhibit certain quality attributes.

2.3.5.4 People Issues

The common issue among the component providers and the component integrators is that of skill requirements.

Component Providers

Designing Reusable Products

The designers of the components have to focus on designing reusable systems that can be easily and safely integrated with other components.

Component Integrators

Evaluating skills and Management Skills

As mentioned earlier, the responsibilities of component integrators is more on evaluating existing components rather than developing new components. They have to compare the existing components to make the necessary trade-offs between cost, component availability, component functionality, quality and integrated system requirements. This requires the integrators to have not only technical skills, but also greater management abilities.

From the above discussion, it can be deduced that the impact of CBA within the alternative industrial domain could be summarised as i) new working practices, ii) changes in work culture, iii) new skills requirements and iv) emergence of new industry standards [52]. It is vital that the managers and stakeholders within the new domains have an appreciation of the level of the impact on their current working practice so that they can evaluate the "paint of change" for their enterprises. Evaluation and detailed understanding is particularly important within the automotive domain due to the conservative approaches to changes in system design and implementations, and stringent time and cost constraints.

2.4 Evaluation

Evaluation is "the systematic collection of information about the activities, characteristics, and outcomes of programmes, personnel, and products in order for interested persons to make judgments about specific aspects of what those programmes, personnel or products are doing and affecting" [53]. The process is usually carried out through a combination of observation, judgment and analysis. Evaluation means different things to different people and takes place in different contexts [54]. The essence of evaluation very much depends on the purpose, the people involved and how it is being used [54-57]. In an evaluation, the object being evaluated is termed as the evaluand [58].

In general, evaluation involves the four Ws and one H: why, what, who, when and how. The 'why' questions aims at asking why an evaluation is required, i.e, the purpose of evaluation. The 'what' attempts to answer what would be evaluated and 'who' directs the question at who are involved in the exercise (who is/are to be studied, who's opinion should be sought). 'When' refers to the timescale of the evaluation i.e. is the evaluation to be carried out before the actual implementation, half way through the implementation or at the end of the implementation. The 'how' is the important aspect of evaluation, how is the evaluation going to be conducted. These five terms outline the strategy to be adopted in any evaluation.

The key components within an evaluation process are illustrated in Figure 2.6. Key issues involved are: i) Planning, ii) Uses (of the evaluation), iii) Evaluation Design, iv) Stakeholders Involvement, v) Data Collection and vi) Analysis and Report. Each of these components is described in the following sections.

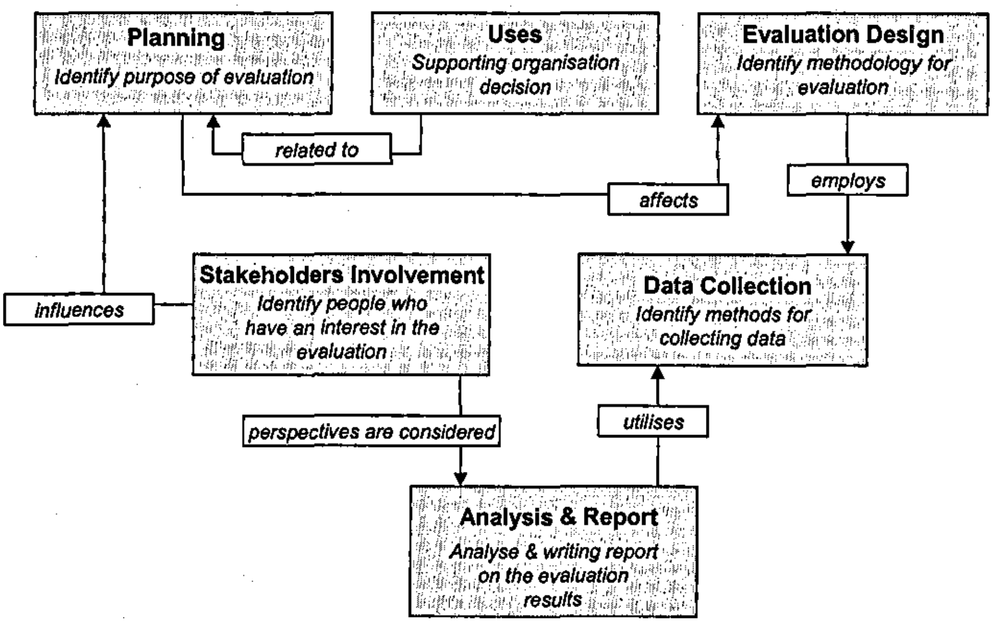


Figure 2.6 The major components within an evaluation

2.4.1 Planning the Evaluation

The planning of evaluation exercise is structured around identifying the purpose of the evaluation. There are many reasons for having an evaluation. Rossi et. al [59] stated that the purposes are determined by asking the questions of : who wants the evaluation, what do they want and why do they want it. They further established the four most common reasons for having an evaluation [59]:

i) Improvement
ii) Accountability

iii) Knowledge generation

iv) Political ruses or public relations

The questions that are usually asked at this stage of the evaluation are: 'why', 'what' and 'who', i.e. "why" is an evaluation necessary, 'who' wants to know 'what' from the evaluation and what is the result going to be used for.

Improvement

The intention of the evaluation may be to help improve the evaluand. In such cases, the emphasis on the evaluation is timeliness, completeness and ensuring that the results are immediately useful. Therefore, it is not uncommon to find that the evaluator will provide regular feedback to the organisation or people who commissioned the evaluation.

Accountability

Evaluations are conducted to judge the merit or worth of an evaluand, assessing if the evaluand has achieved its objectives and its performance requirements. This is particularly applicable to a programme, such as social programmes and research programmes. Very often, the findings of these evaluation are intended for the decision and policy makers, and may influence the programme continuation, allocation of resources, restructuring or legal action [60]. As such, evaluation conducted requires sufficient credibility to provide a basis for action and criticism.

Knowledge Generation

Evaluation could also be intended to generate knowledge about the evaluand. The evaluation findings may not be used for any decision makings but just to understand the effect of intervention to the evaluand. The findings contribute by increasing knowledge [61] and are disseminated publicly, such as through academic journals, conferences and other professional outlets. This knowledge could be used by others to design and shape other programmes or work processes.

Political Ruses or Public Relations

Rossi et. al. [59] pointed out that certain evaluations are conducted not for performance assessment but purely for publicity effects, such as to impress funding organisations or political decision-makers. Under such circumstances, it is important to clarify the nature of the evaluation, identify appropriate and realistic expectations and direct the effort toward appropriate uses [53, 62].

2.4.2 Uses of Evaluation

The use of evaluation findings is very much related to the purpose of evaluation. It is therefore common to find that the findings are used for decision making, identifying areas for improvement and knowledge generation (see section 2.4.1).

Patton [61] has introduced the notion of "process use" to the uses of evaluation findings. Process use refers to the learning of the individual through their engagement in the evaluation process, which eventually leads to changes in thinking, behaviour and changes to the programme. That is to say, on top of the evaluation results, individuals involved in the evaluation process learn something new about themselves or about the programme, which they might not have been able to without the evaluation. Such learning could be for the evaluator, individual involved in the programme being evaluated or the organisation involved in the programme.

2.4.3 Evaluation Design

The evaluation design is a logical model that is used to gather evidence on results that can be attributed to the evaluand [63]. The common types of evaluation design are: surveys, case studies and experiments. A differentiation has to be made between evaluation design and data collection methods. Evaluation design is about the *methodology* for the evaluation and many different methods of data collection could be used within the methodology.

Experimental Design

Experiments isolate individual factors and observe their effect in detail. The purpose of experiments is to discover new relationships or properties associated with the subjects being studied or to verify existing theories [64]. The factors under study are called the variables. Variables which are the input parameters and manipulated by the experimenter are called the independent variables; variables that are the outcome, effect or output of the input variables are called the dependent variables [65]. A change in the independent variables affects the dependent variables but changes in dependent variables does not affect the independent variables. Experiments involved the manipulation of circumstances, the identification of causal factors and observation and measurement [66]. Groups are usually set up for comparison, one experimental group which is subjected to the stimuli under investigation and a comparison group which receives no stimuli. The groups are selected through random assignment. The outcomes and changes that occur are observed and measured in detail and precision. Hence, experiments identify causal relationships. Causality is shown by observing associations between the stimuli and the dependent variable [67].

Experimental designs are often regarded as the most rigorous approach for evaluation design to establish a causal relationship between an evaluand and the evaluation results [63, 68]. This is based on the fact that if two groups of experimental units, for example two groups of people are "equivalent" to each other (i.e. similar people, live in similar context, having same backgrounds), one group is given treatment and the other is not, then if differences are observed between the two groups, it is logical that the differences has to be due to the treatment one group received and the other did not. Because the groups are randomly selected and assigned from a common group of people, bias in selection is reduced.

Experiments can be conducted in laboratory or in the real world. Laboratory experiments are set up to implement specific measures in a controlled setting. Such evaluations will be designed to collect objective measurements (such as time to complete a task), gather information on responses and errors. The evaluator will have to specify the questions to be answered and designed the task for the experiments. The method of data analysis is often statistical analysis [69]. The advantage of controlled experiments is that objective measurements can be obtained. The evaluators can observe the test participants during the test and gain valid feedback. The disadvantages of these experiments are that they are time-consuming and relatively expensive to conduct.

Experiments carried outside a laboratory setting are termed as real world experiments and example of these are field experiments. In such experiments, the researchers are unable to manipulate and dictate the circumstances; some of the variables are not within the control of the researchers.

Quasi experiments are a type of experimental design in which there is no random assignment of groups, rather the groups are selected based on similarity. Quasi experiments are considered "inferior" to "standard" experiments because without random assignment, one can never be sure that the groups are equivalent [68].

In non-experimental design there is no treatment group or comparison group. Because no treatment group is set up, these experiments cannot predict the causal relationships [70]. These experiments study the phenomenon and no attempt is made to change the subject of the research [71].

Tests (in contrast to experiments) are commonly used, such as e.g. driving tests, eye tests, aptitude tests, and abilities test. The processes behind tests are similar: questions are set and answers are given; tasks are requested and actions followed. Tests share some similarities with experiments [64]. Firstly, tests have a uniform procedure, giving the same questions or tasks to all people. They are administered within a controlled environment. Secondly, tests measure a specific trait and often require observation and measurements. Tests involved classifying, categorising and enumerating specific traits (such as people's skill or knowledge), with the aim of measuring such traits in terms of numbers (e.g. 5 out of 10) or categories (e.g. useful or not useful). Thirdly, comparison is generally made with respect to "universal". For example, so called "standardised" tests have been thoroughly assessed with respect to a representative sample of the appropriate population. These tests have established their reliability and could be used as objective benchmarks for comparison [64].

Survey Design

Surveys are studies where a sample of a larger population is selected in order to draw inferences about that population because the resources are not available to study every member of the population [72], i.e. sampling. Each survey pertains to a given point in time and aims to incorporate as wide and as inclusive a data set as possible. Very often, surveys relate to the present state of affairs and attempt to provide a snapshot of how things are at the specific time when the data are collected [73].

Generally, standardised information is collected from the sampling population by asking the respondents a same set of questions. There is no attempt to manipulate variables or control conditions. Surveys are used when the interest is to seek opinions, study attitudes and values, explore aspects of a situation or to seek explanation. This helps to provide a quantitative picture of the individuals [69]. Surveys are a way of collecting primary and quantitative data[3]. They are a correlational method and thus do not establish a causal relationship [74]. For example, it might be found that people who eat breakfast are less likely to become obese, but it cannot be said that breakfast does not cause obesity.

Surveys can be carried out over a short period of time or over a long period i.e. longitudinal research in which the same set of people, issue or situation is studied over a period of time. They can be conducted using different methods: postal questionnaire, face-to face interviews, telephone interviews, documents and through observations [66].

The advantages of surveys are that empirical data collected, i.e. the data are based on the respondent's own experience rather than on theory and the data are collected directly from the respondents [64]. Surveys often have a wide and inclusive coverage and are more likely to produce data that can be generalised to be representative of the whole population. Surveys are also able to generate a large amount of data in a short time at a fairly low cost.

One of the disadvantages of surveys is that the data are affected by the characteristics of the respondents who could not necessarily be truthful in their responses. Accuracy of the responses cannot be checked in surveys although consistency can be monitored by appropriate structuring of the questions [75].

[3] Primary and quantitative data will be explained later in Section 2.4.5

Case Studies Design

Case studies focus on one or few instances of a particular phenomenon with a view to providing an in-depth account of events, relationships, experiences or processes occurring in that particular instance [64]. Often in case studies, one or a few examples are selected from a wider range of examples, so as to study the example in greater detail. The selected case (or cases) is not a sample, it is studied in its context. The result from the case could be used for analytical generalisation but not expandable to the populations at large [76].

In case studies, a variety of sources, data and methods could be used as part of the investigation. They are empirical in the sense that they rely on the collection of evidence about what is going on. The detailed information enables the establishment of a causal relationship or processes [77] within each case.

The major advantage of adopting case studies as a strategy is that the focus on one or a few examples allows the researcher to study the processes and the relationships in detail. The use of multi-methods and multi-sources of data facilitates validation of data through cross-referencing. It is particularly useful if the researcher has no control over events since the approach is concerned with investigating phenomena as they naturally occur [78].

2.4.3.1 Selecting an Evaluation Design

Within any evaluation exercise, several evaluation designs could be used to obtain different results. For example, in trying to understand a lunch service provided to the elderly within a community, an evaluator may use the survey technique to obtain the profile of the beneficiaries and to seek their opinions on the programme. However, if the evaluator would like to conduct an in-depth study of an elderly who benefited from the programme, then case studies could be used. To compare the effect of the programme with those who received the service and the effect of not receiving the service, an experiment could be designed to study this effect.

There are two major factors to consider when deciding which evaluation design approach to take, namely: i) the purpose of the evaluation and ii) type of questions that the evaluation seek to answer. Yin [76] has provided a guide to decide on which approach to adopt based on (i) the type of questions to be asked i.e. "who", "what", "where", "how" and "why", (ii) would control over events be required and (iii) does the evaluation focus on contemporary events as opposed to historical events (Table 2.2).

Approach	Form of evaluation questions	Requires control over behavioural events ?	Focuses on Contemporary events?
Experiments	How, why	Yes	Yes
Survey	Who, what, where, how many, how much	No	Yes
Case studies	How, why	No	Yes

Table 2.2 Types of questions in relation to the evaluation approaches (Source: adapted from Yin 1984 [42])

It is important to note that the "who", "what", "why" and "how" questions in this section are different from those listed in Section 2.4 which are related to evaluation in general. The three Ws under the context of evaluation design go a step further by relating these questions to the context of the evaluand. For example using the lunch service example again, the "what" questions would be what are the ways in which lunch is provided, what is provided for lunch, what are the criteria for selection. As described by Yin [76], the "what" has another dimension which refers to "how many" or "how much" such as how many people are involved in the programme, what is the amount spent on each participant. The "who" questions would be: for whom is the service targeted, who is delivering the service; and the "why": why should the lunch service be provided, why are some elderly not participating.

Therefore, from the classification in Table 2.2, when the evaluation questions tend towards finding how and why and the evaluator wants to have control over events and focuses on contemporary events, experimental design is preferred. Survey design is used when the evaluation questions tend towards identifying who, what, where, and finding out how many or how much. In addition surveys could be focused on contemporary events but do not required control over the events. Case studies are best suited in finding out the "how" and "why" in an evaluation. The evaluator does not require control over the events and the events are contemporary.

2.4.4 Stakeholders Involvement

Stakeholders are the individuals, groups or organisations that have an interest in the subject under evaluation. For example, the people who provide the funding, i.e. the sponsors, for a certain programme, the people with decision making authority over the programme, the administrators and personnel running the programme and the intended beneficiaries or people affected by the programme. Identifying the stakeholders is a crucial part of the planning of an evaluation because the purpose of the evaluation is dependent on who wants the evaluation, what they want and why they want it (see section 2.4.1), and who are the ones affected by the programme [10, 60, 62, 79]. Stakeholders are the groups at risk and have a stake in the evaluand [51]. They ought to have the opportunity to make their claims, their concerns and raise questions, and to have their input in the evaluation.

Different stakeholders have different levels of risk to bear and consequently must be able to feel that they have some defence against that level of risk. On the other hand, stakeholders are also the users of the information from the evaluation [62]. They will use the information to address their claims, concerns and issues. It is important that the evaluation takes into consideration their views and addresses the topic from their perspectives.

The stakeholders have a good understanding of particular aspects of the evaluand and can provide the evaluator with the practical and factual knowledge to enable the evaluator to have a good understanding of the evaluand [80]. This will increase the validity of the findings. It has been found that for an evaluation to be successful and to achieve wide credibility it is important that it does not focus solely on one viewpoint, it has to take in the views of different stakeholders [10]. This will increase the chances of the evaluation findings to be accepted, recognised and used.

2.4.5 Data Collection Methods

Evaluation design is not the major determining factor for the data collection method. As mentioned earlier, different methods of data collection could be used within each evaluation design. It is the nature of the data required and the sources available that are the major influences in selecting the method to be adopted [63]. The data required, nevertheless depends upon: i) the evaluation design, ii) the indicators that is required for capturing the results and iii) the type of analysis to be conducted.

Data can be classified as either quantitative or qualitative [81]. Examples of quantitative data are numerical observations, such as e.g. hours, distance, number of pages and lines. These data can be quantified in numerical terms. Qualitative data, on the other hand, are data that cannot be quantified in numerical terms (e.g. words or images, the product of a process of interpretation or the result of the way they are interpreted and used by researchers) [64].

Data can be further classified as subjective or objective. Subjective data rely on personal feelings, attitudes, experience and perceptions. Objective data are observations based on facts that in theory, involve no personal judgement.

Data can also be classified by their source i.e. primary or secondary. Primary data are collected by the researcher directly at the source whilest secondary data have been collected and recorded by another person or organisations.

The common data collection methods are questionnaires, interviews, observation and analysis of documents and are explained in more detail below.

Questionnaires

The questionnaire is a common data collection method that can be used to collect primary data and quantitative or qualitative data [82]. Usually, a list of questions that the evaluation seeks to address is developed. The questions and responses can be either closed or open ended. Closed ended questions provide the respondents with fixed responses to choose from. For example, what do you think of the programme, very bad, bad, average, good or very good. The respondents may be asked to indicate their opinion (agree or disagree with a statement) or provide rating to a statement. Very often, closed ended questions tend towards providing quantitative data. The evaluator would be able to quantify the number of people who share the same opinion. Open ended questions, on the other hand, require the respondents to express their personal feelings freely, such questions tend to provide qualitative data. Research has found that open ended questions in a questionnaire could be used to substantiate the results from the close-ended questions [83].

Questionnaires often use codes to identify particular responses or types of responses. For example, the response to a question about the respondent's gender might be coded as '1' for male and '2' for female. Such numbering of the answers helps to organise, quantify and facilitate data analysis [84].

Questionnaires can be administrated face-to-face with the respondent or they can be posted to the sample population (i.e. self-administrated). They can be the easiest and least expensive method for obtaining measurement data, depending on the type of data to be collected. However, the questions have to be specific in order for the respondents to provide a valid opinion. In cases where self-administrated questionnaires are used frankness and honesty to the questions are encouraged because of anonymity [74]. Questionnaires are relatively easier to arrange than other methods, for example, a postal questionnaire can be used to reach out to a larger population than an interview.

The downside of questionnaires is that the questions are structured and may shape the nature of the responses. Another disadvantage is that there is often a low response rate to questionnaires depending on the research topic, the sample group, the interviewer's appearance in a face-to-face interview and the social climate [64, 67, 74]. It is also difficult to check the truthfulness of answers in a questionnaire.

Interviews

Interviews involve discussion between a researcher and a person who provides data that are relevant to the research. Interviews can be highly structured with pre-designed questions or completely unstructured in which the interviewer just starts a topic and allows the interviewee to develop his or her idea and speaks freely without interruption. The interviews can be one-to-one or group interview in which the interview is conducted with several people at the same time; it can be face-to-face interviews, telephone interviews or written form [85].

Interviews are straightforward ways of finding out information. They are a way of collecting primary data and very often the data collected are qualitative. The data from an interview will consist of direct quotations from people about their experiences, opinions, feelings and knowledge [86]. The results provide qualitative insight into the individual, personal experiences of the interviewee. Response rates are often not issues of concern since interviews are generally pre-arranged and scheduled. They also allow opportunities for probing and clarification, finding out why people response the way they did.

Interviews are time-consuming since only one person or one group can be interviewed at one time effectively. If several people are interviewed at once, some people may go along with the group rather than give their true opinions. The data derived from interviews are generally open ended, making it difficult for analysis. In addition, the data collected are to an extent unique to the specific context and the specific individuals involved and this has an adverse effect on reliability of the data [64]. Interviews are also subjected to interviewer bias; the interviewer may influence the respondents by the way the questions are phrased and by verbally or non-verbally encouraging “correct” answers.

Observation

Observation (sometimes referred to as protocol analysis [87]) is an evaluation method in which the researcher observes the activities of the people being evaluated, their actions, behaviours, interpersonal interactions and organisational processes. However, the data collected is often being associated with an event or other occurrences rather than people [88].

There are two types of observation: naturalistic observation and participant observation. In the former, the evaluator adopts a detached attitude, observing the subject under study unobtrusively. In participant observation, the evaluator become part of the observed group, interacting actively with the respondents, i.e. fieldwork, collecting data in real life situations at first hand. Such fieldwork requires the researcher to be with the observed group for a period of time. The data collected from such observations are primary and qualitative, they have depth and detail; they must be descriptive so that reader can understand what occurred and how it occurred. The descriptions must be factual, accurate and thorough [83].

Though observations tend to be associated with qualitative data, they can also be used for collecting quantitative data. Such data tends to be from structured observations. In a structured observation, observation guidelines or a checklist is created to provide a guide for the observer to record the proceedings of the observation. These observations can be based on frequencies of events, duration of events, events at a given point in time (for example logging down what happens at an interval of every 25 seconds), or sample of people (for example, an individual is observed for a pre-determined period of time after which the observer will switch his attention to observe another person) [64].

Robson [69] has proposed that the major advantage of an observation is its directness. The researcher does not ask the people about their views, feelings or attitudes; rather, the researcher just watches what they do and listens to what they say. The method also provides a rich insight to the research topic. However, the disadvantages are: i) observation is time-consuming, ii) an observer might have an influence on the situation under observation and iii) observations are dependent on the researcher's skills and experience which is a source for doubting the reliability of the data and how representative the data are [77].

Analysis of Documents

Analysing documents as an evaluation method essentially involves using various kinds of documents as the key source of data. The documents provide background information for the evaluation (i.e. literature research) and also a source of data. Documents may come in the form of written sources, visual sources and audio sources [89]. These could range from books, journals, newspapers, web pages, video footage, photographs, letters, memos, diaries, government publications and official statistics. The main concern is the accessibility of documents.

The advantages of using documents as a source for evaluation is that there are vast amounts of permanent information held in documents . However, the downside is that documents are often secondary data that may have been produced for other purposes. Furthermore it is also for the researcher to evaluate the credibility of the source that produced the data. However documents are economical and often an easily accessible source of background information that can shed light on the real scope of the problem or help familiarize the researcher with the situation and the concepts that require further study [88].

2.4.6 Analysis and Reporting of Data

The final stage of the evaluation process is the analysis and the reporting of the data. As stated above (section 2.4.5), the data are either quantitative or qualitative. As Denscombe [64] has observed, the most elementary distinction between quantitative and qualitative data is that the former are in numbers and the latter are in words. For quantitative data, depending on the size of the data, they could be either processed manually or by using analysis software (e.g. Statistical Package for the Social Sciences i.e. SPSS, Minitab). If the data are already in numbers, it will be ready for analysis. They might need to be grouped and presented in terms of graphs or charts. These data might be further subjected to statistical analysis, which will enable the evaluator to move beyond individual interpretation by using scientifically based tests. Statistical test are often used when: i) describing the frequencies of occurrences and their distribution, and ii) looking for connections between variables or categories in the data [64].

For qualitative data, the raw data has to be processed and organised before analysis can be undertaken. For example, taped interviews have to be transcribed, field notes or documents have to be organised or categorised. Patterns and processes, commonalities and differences [86] are the main things to look out for during the analysis. During the process, the evaluator might need to go through notes taken during data collection repeatedly to look for themes or interconnections. If necessary, the evaluator might have to revisit the field (such as confirming the findings with interviewee) to check the validity of the themes or interconnections. Finally, the evaluator should aim to refine a set of generalisations that explain the themes and relationships identified in the data [64, 69].

Lastly, the analysis has to be written and produced as a formal record of the work.

2.5 Enterprise Modelling

Modelling is an essential part of the evaluation process, especially when the stakeholders need to appreciate the effects of situations and / or new technology that is beyond their current experience. For manufacturing enterprises, static and dynamic "enterprise modelling" can provide vital input to the understanding and analysis of the issues that might be encountered.

According to Vernadat [90], enterprise modelling is the set of activities or processes that can be used to develop the various parts of an enterprise model. Enterprise models are the abstraction of real life systems [91]. They are created to "assist an analyst extract requisite details of a system, in order to gain a better understanding of the complexity"[91]. This is because models attempt to simplify the system in order to reduce the complexity, bringing clarity and understanding to some aspects of the system where there is complexity, changes or assumptions. Therefore, a model always has a scope, purpose and perspectives (or foci of concern such as financial, information and functions views), which may be explicit or implicit [44].

An enterprise model is a symbolic representation of the enterprise, containing individual facts, objects and relationships that occur within the enterprise [91] i.e. they are abstractions of how the enterprise conducts its business [92]. Enterprise modelling attempts to answer the question of the "what, how, when and who" aspects of the enterprise [90]. They provide a structural representation and enable the analysis of the activities and their interactions [91].

Enterprise models help to represent and enhance understanding of how an enterprise works by using useful representations determined by the modelling constructs (i.e. the syntax and semantics of the modelling language, which could be graphical symbols, textual statements, or logic and mathematic expressions). With this knowledge, enterprise modelling can be used for [90]:

i) capitalising acquired knowledge and know-how for later reuse,
ii) rationalising and securing information flows,
iii) designing and specifying a part of enterprise,
iv) analysing some aspects of enterprise,

v) simulating behaviour of parts of enterprise,

vi) making better decisions about enterprise operations and organisation, and

vii) controlling, coordinating or monitoring certain parts of the enterprise. [4]

Due to the complexity and size of an enterprise, an enterprise can be represented by a set of models. Each model covers a part or subset of the enterprise, representing some detailed aspects of the enterprise from a given perspective [90] such as, behaviour [93], function [94], human resource modelling [95]and information modelling [96]. The goal is not to represent the entire enterprise but to focus on a part of the company that needs to be represented as defined by the stakeholders.

Enterprise models may be static or dynamic. According to Veryard [44], a static model is a snapshot of the enterprise or organization at a particular time, which may include its structure, boundaries with the environment, processes, strategic objectives, competitiveness, values, external influences and a SWOT (strengths, weaknesses, opportunities & threats) analysis. A dynamic model is one that represents the way these things change over time [97].

Enterprise modelling is often associated with enterprise engineering [98]. Enterprise engineering involves the engineering of an enterprise in a systematic way, identifying the need for changes in enterprises and the implementation of changes expediently and professionally [99]. In the highly competitive global economy, enterprises are subjected to constant challenges. Businesses need to renew themselves constantly. This process of renewal needs continuous and efficient design and redesign of the enterprise (i.e. engineering) so that it can keep up with the competition [100]. The ultimate aim of enterprise engineering is to enable an enterprise to achieve its business goals, i.e. be cost-effective and be competitive in its market environment [90]. This is achieved by improving the engineering of the communication infrastructure between enterprise elements involved in the enterprise operation (such as people, machines) and their coordination and cooperation [98]. This requires the identification of the relevant information to support the engineering activities.

[4] Item (i), (ii), (iv), (v) and (vi) have been used in the research work reported in this thesis. A detailed description will be given in Chapter 6.

In recent years, business process re-engineering (BPR) has gained much interest [101]. BPR focuses on the review and redesign of the core processes of an enterprise. Enterprise engineering, on the other hand, is the review and redesign of the entire organisation, in which BPR is one of the aspects. The role enterprise modelling plays in enterprise engineering is to capture the business processes of the enterprise, enabling a thorough understanding, engineering and control of the business [102]. This enables the capturing and presentation of the relevant information required for redesigning an effective and efficient enterprise.

2.5.1 Process Modelling

Process modelling has been extensively studied in software engineering [103]. In software development, process modelling is a description of the activities involved in the software development and maintenance life cycle. This implies that a process model is a representation of the activities. Vernadat [90] adapted the idea of process modelling to enterprise modelling. This is referred to as process-based modelling, in which the process refers to business process.

Business process models describe both the functionalities and the operational behaviour (dynamics) of the enterprise and identify all the required and generated information [98]. They are representations of the processes, capturing the relationships with the internal and external environments.

By using modelling techniques, it is possible to represent the business processes through which work is implemented and accomplished. Each process has a supplier providing the inputs and a customer to receive the outputs. The information that is being offered from a process model are: what is being done, who does it, how it is done, who or what is dependent on its being [103]. Graphical representations of processes are considered to be easily understood and comparable with each other [104].

Process models can be used to support decision making at any level of the enterprise organisation for strategic, tactical or operational planning [104, 105]. They also promote a common understanding about the content of the operational processes amongst the stakeholders [98]. Process models prove to be useful when it comes to supporting exception handling [98]. The models can be used to simulate proposed corrective process changes and to evaluate their impact on the process itself as well as on the overall operation.

2.5.2 Dynamic Modelling

The concept of dynamic modelling is best explained by contrasting with static modelling. Static models capture a snapshot of something, for example a system at a particular point in time [106]. The model does not change as time evolves. Dynamic modelling, on the other hand, attempts to express and model the behaviour and interactions of the system over time, providing a view of the evolution or the system [97, 106]. By this definition, dynamic modelling can also be referred to as simulation [106]. The dynamic modelling of business processes is also referred to as business process simulation which captures the dynamic behaviour of a process [107].

As mentioned earlier, an enterprise model can be static or dynamic; dynamic enterprise models are inherently more complex than the static models because they attempt to show changes over time. They are able to provide more information, such as providing "what if" analysis. This does not discredit the function of a static model. A static enterprise model, for example, provides a graphical description of the enterprise, capturing the functionality and relationships among the processes and resources, for the purpose of communicating and understanding by the stakeholders. A dynamic model would be able to provide information on changes. Additionally they can be used to support exception handling [98] and decision making. Alternative solutions could be simulated through dynamic modelling and the impact of changes to the various aspects of the process could be evaluated before any changes are implemented.

2.5.3 Enterprise Reference Model

The modelling of enterprises is facilitated by the adoption of a reference architecture and a method, so as to i) manage the inherent complexity and dynamic nature of an enterprise, ii) make sure that an desired model could be obtained, iii) obtain a valid representation of the reality, and iv) ensure a formal structured representation of events, processes and resources is adopted [90, 108]. The following sections will briefly discuss the "state-of the art" for reference architectures for enterprise modelling and the associated modelling approaches. In general, the different reference architectures cover the major aspects of a system that need to be addressed although each approach emphasises different aspects of the system. The main focus in what follows is on the CIMOSA reference architecture since (i) this has been used as the basis for the enterprise modelling and evaluation used in this thesis and (ii) there is extensive previous experience of its constructs and usage in the author's research group (i.e. MSI Research Institute).

CIMOSA

The Computer Integrated Manufacturing – Open System Architecture (CIMOSA) was developed by the AMICE Consortium during a series of ESPRIT Projects[109]. CIMOSA aims to help companies manage changes and integrate their facilities and operations to enable competition on price, quality and delivery time. CIMOSA introduced an event driven, process-based approach to integrated enterprise modelling, ignoring organisational boundaries as opposed to function or activity-based approaches. CIMOSA adopts the idea of open system architectures for CIM, such that an enterprise is comprised of vendor-independent, standardised CIM models, described in terms of their function, information, resource and organisational aspects. The models are designed according to a structured engineering approach that can be plugged into a consistent, modular and evolutionary architecture for operational use [90]. CIMOSA is founded on a modelling framework that is based on a *reference architecture* from which a *particular architecture* can be developed for specific application domains. The framework is usually represented as a three dimensional cube, offering the ability to model from orthogonal aspects and views of an enterprise [109], namely:

- a genericity dimension concerned with the degree of particularisation. Genericity ranges from generic building blocks to a particular model.
- a modelling dimension which provides the modelling lifecycle support from initial system specification and requirements definition to a description of system implementation.
- a view dimension concerned with system behaviour and functionality which enables the development of sub-models representing different aspects of the enterprise such as function, information, resource and organisation.

Figure 2.7 shows the conceptual modelling framework of CIMOSA. The CIMOSA architecture (i.e. reference and particular architecture in the diagram) and its modelling approach (i.e. view and modelling dimensions) will be discussed in greater detail in Chapter 6 of this thesis.

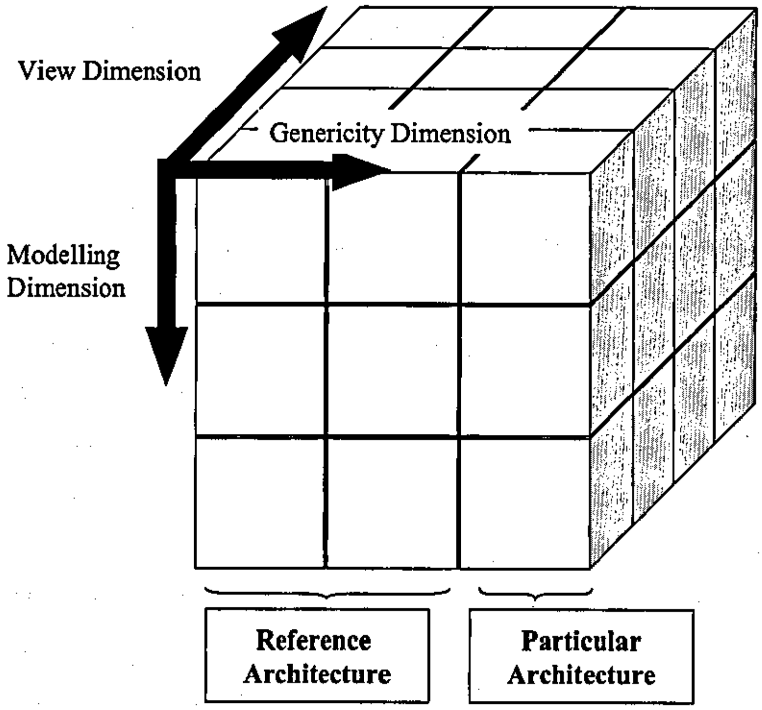


Figure 2.7 CIMSOA modelling framework (adapted from AMICE 1993 [109])

ARIS

The ARchitecture for integrated Information Systems (ARIS) deals with business-oriented aspects of enterprises (such as order processing, production planning and control, inventory control), focusing on software engineering and organisational aspects of integrated enterprise system design [90]. ARIS's structure is similar to CIMOSA, having four views and three modelling levels. The four views cover function, data, organisation and control. The three modelling levels are the same as those of CIMOSA: generic, partial and particular.

The function view defines a hierarchy of functions in terms of structograms, program modules and program flow from conceptual and technical models through to implementation. The data view defines the semantic entity-relationship diagrams (i.e. conceptual) and enables the translation into relational schemata (i.e. technical) prior to embodiment within physical database system (i.e. implementation). The organisation view defines the enterprise structure by the use of a standard organisation chart (i.e. conceptual), the network topology (i.e. technical) and the physical network (i.e. implementation). The control view presents the business processes and their implementation as logical sequences of program execution with relevant computer screens and distribution of data over the enterprise network. A graphical representation of the architecture of ARIS is given in Figure 2.8.

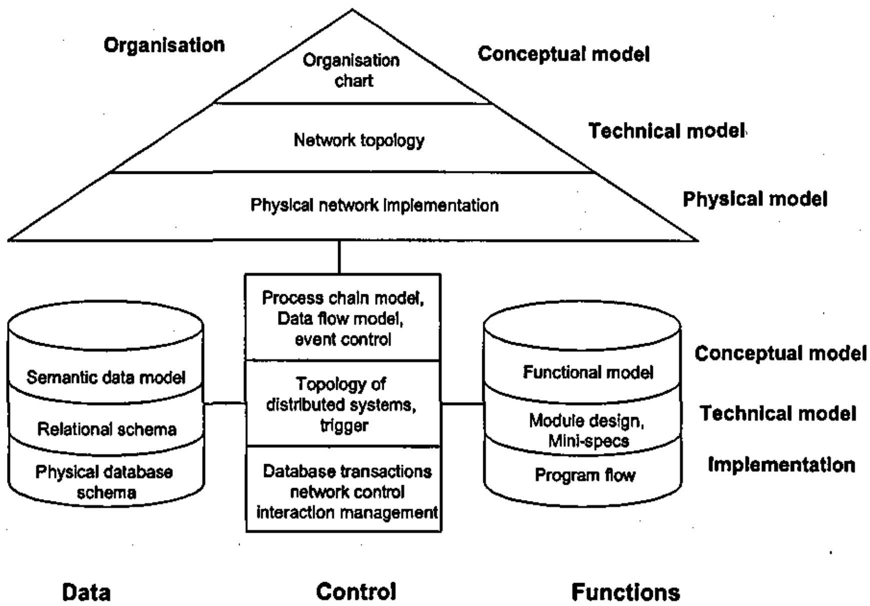


Figure 2.8 The ARIS architecture (adapted from Williams and Rathwell [110])

GRAI/GIM

The Graphs with Results and Activities Interrelated / GRAI Integrated Methodology was developed at the University of Bordeaux in France [111]. It is a methodology for conceptual design and analysis of manufacturing systems. The scope of GRAI-GIM covers the conceptual, organisational and physical aspects of a system. The conceptual model was borrowed from general systems theory and systems organisation theory [90]. Figure 2.9 shows a graphical representation of this conceptual model. GRAI/ GIM identifies four sub-systems within an enterprise: physical system, information system, decision-making system and an operating system. Each of the sub-systems can be decomposed into more detailed physical, information and decision-making systems. It follows that in GRAI-GIM an enterprise can be described from four views: information, decision-making, physical and functional. The methodology is based on three abstraction levels: conceptual, structural and realisation. In addition, GRAI-GIM employs a user-oriented method and a technically-oriented method. The user-oriented method is used to express user requirements specifications in terms of function, information, decision-making and physical systems. The technology-oriented method defines requirements in terms of organisation, information technology and manufacturing technology [112].

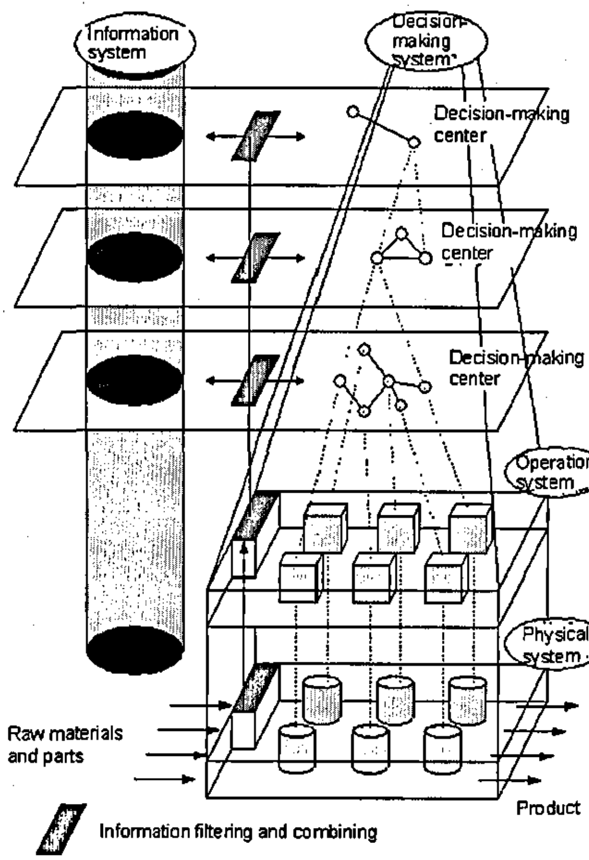


Figure 2.9 The GRAI conceptual model (Source: IAST, [113])

PERA

The Purdue Enterprise Reference Architecture (PERA) is characterised by its layering structure [90]. It has been created to cover the full enterprise life-cycle from conception and mission definition down to the operational level and final plant obsolescence (see Figure 2.10). Each layer defines a task phase and each task phase is described by a technical document containing a set of procedures which will lead a user's application group through all the phases of an enterprise integration program [114]. PERA is not a modelling method and as such does not provide its own modelling language. Other modelling tools and techniques can be used to support its enterprise modelling concepts. PERA identifies three main classes of entities of an enterprise, information, human and manufacturing equipment. In contrast to the alternative reference architectures, PERA places particular emphasis on the human aspects within the organisation [113].

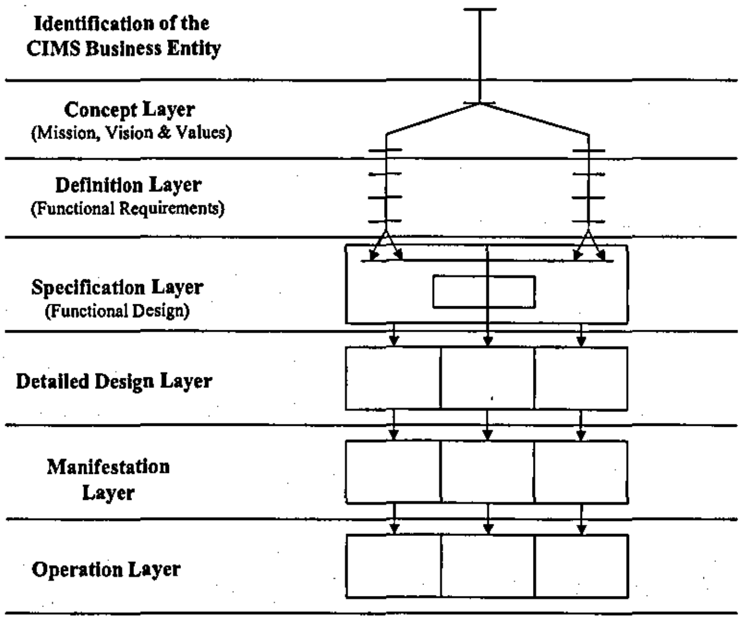


Figure 2.10 PERA skeleton (adapted from Vernadat 1996, [90])

GERAM

The Generic Enterprise Reference Architecture (GERAM) has been developed to serve as a reference for the whole enterprise integration community. GERAM builds on the results from CIMOSA, GIM and PERA and as such can be considered as a generalisation of existing architectures and other necessary elements [90]. The architecture attempts to provide and / or promote: i) definitions of terminology, ii) a consistent modelling environment, iii) a detailed methodology, iv) good engineering practice for building reusable, tested and standard models, and v) a unifying perspective for products, processes, management, enterprise development and strategic management [115]. GERAM defines a tool-kit of concepts for designing and maintaining enterprises for their entire life-cycle [100]. It has nine major components: i) Generic Enterprise Reference Architecture (GERA), ii) Enterprise Engineering Methodology (EEM), iii) Enterprise Modelling Languages (EML), iv) Enterprise Engineering Tools (EET), v) Enterprise Models (EMs), vi) Enterprise Modules (EMOs), vii) Generic Enterprise Modelling Concepts (GEMC), viii) Partial Enterprise

Models (PEM) and xv) Enterprise Operational Systems (EOS). The major components of GERAM and their relationship are summarised in Figure 2.11. Although GERAM would appear to be the most complete reference architecture and hence the one to be adopted in any application it is the complexity associated with this all-encompassing approach that has limited its exploitation.

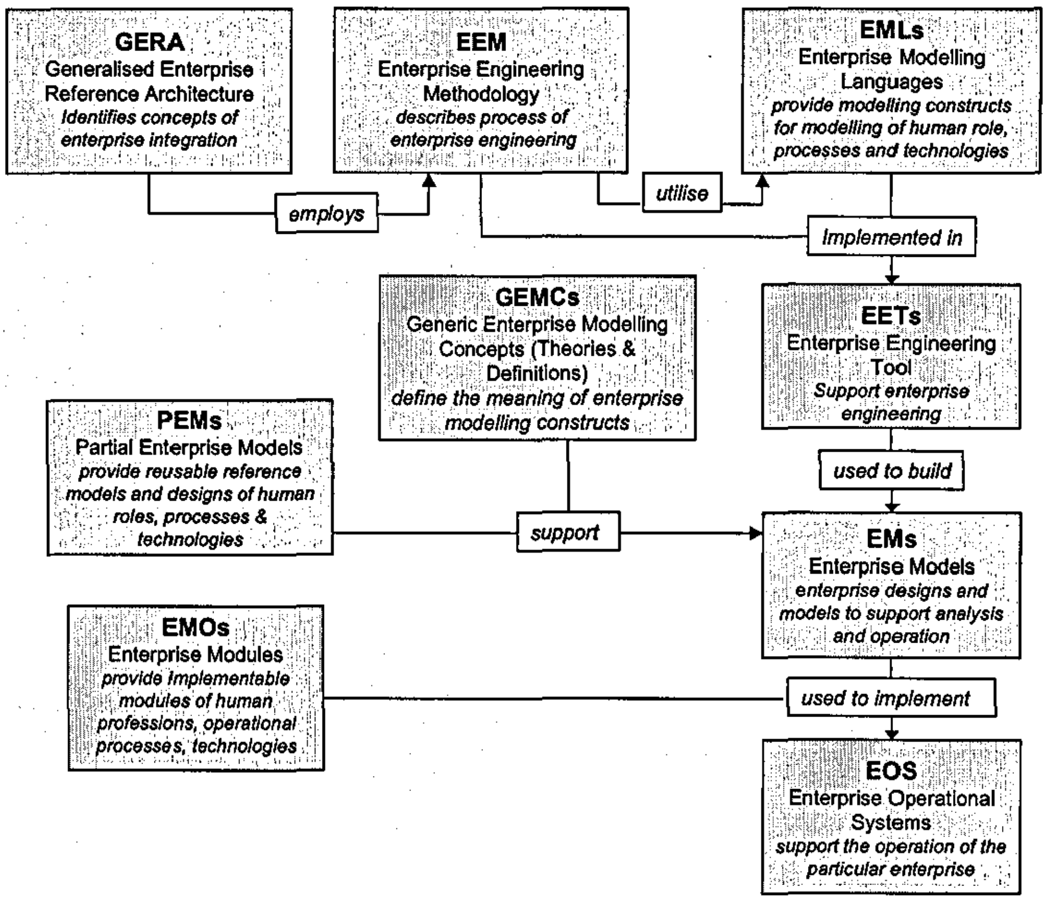


Figure 2.11 The framework and the major components of GERAM (adapted from GERAM 1999, [115])

2.5.4 Modelling Methods

Reference architectures ensure that relevant areas of concern are addressed when an enterprise modelling activity is undertaken. In some respects they can be considered as "route maps" that ensure the consistency of the modelling approach adopted. Within these route maps an appropriate enterprise modelling method needs to be selected to enable the definition and implementation of enterprise models. There are a number of competing methods available e.g. IDEF, SADT, CIMOSA and IEM, which are described below.

IDEF Suite of Modelling Method

The IDEF modelling approach is based around a suite of modelling methods [91]. Each of the methods in the suite provides a set of modelling syntax and steps for describing a particular perspective of an enterprise. The suite provides for functional modelling (IDEF0), information modelling (IDEF1), data modelling (IDEF1x), system dynamics modelling (IDEF2), process description capture (IDEF3), object-oriented design (IDEF4), and ontology capture (IDEF5). IDEF3 and IDEF5 are descriptive, while the rest can be used to build a model [110].

The advantages of the IDEF method are that it is simple and easy to understand and it is supported by a number of commercially available computer-based tools [90]. The disadvantages of IDEF are that it results in: i) static (not computer processable) models of systems, ii) ill-defined system behaviour, iii) models that cannot be integrated and iv) the same concepts needing to be repeated in different models. These disadvantages give rise to poor consistency checking capability. It is limited for the purpose of detailed process design and validation at the engineering level [90].

SADT

SADT (Structured Analysis and Design Technique) is a technique for system planning, requirements analysis and system design [116]. SADT makes use of two types of models: activity models and data models (referred to as the actigram and datagram respectively). A SADT activity model describes the decomposition of activities. The SADT data model describes the decomposition of data. Each SADT diagram is composed of boxes (representing activities) and connected by arrows (representing flows of materials, data or information). The functional decomposition principle in SADT and the graphical notation of SADT is shown in Figure 2.12. Although there are advantages in adopting the SADT (e.g. a modular structured functional decomposition, simple graphical representations) the issues associated with dynamic modelling (i.e. problems in specifying discrete event dynamic systems because of an inability to handle resource and control flows and a lack of semantic explanation) limit its applicability for many enterprise modelling activities [90, 105].

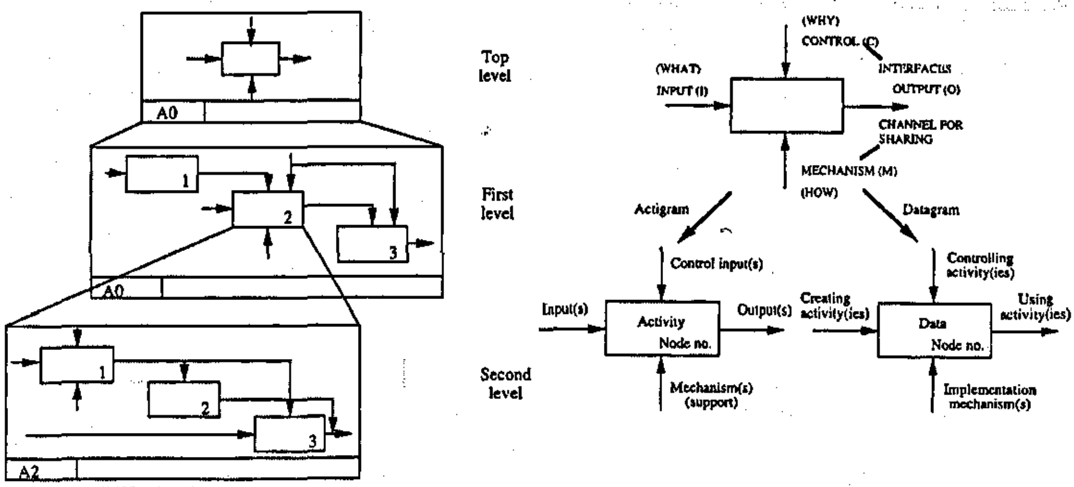


Figure 2.12 Functional decomposition principle of SADT is shown here on the left, and on the right is the graphical notation of SADT (Source: Vernadat 1996, [90]).

CIMOSA

Within the CIMOSA reference architecture is the provision of a set of modelling structuring concepts and constructs that have been defined in terms of object classes and elements [117] (see Figure 2.13). CIMOSA employs the object-oriented concepts of inheritance to structure its constructs into a hierarchy of object classes covering the meta, object and elemental levels (see Figure 2.13). The object classes are organised into generic building blocks and building block types. The generic building blocks define the set of basic primitives of the modelling language provided to the users. Building block types are specialisations of generic building blocks, i.e. more specific sub-classes of basic primitives (using the property inheritance principle as found in any object-oriented language) covering, for example, domains and business process events, enterprise activities, enterprise objects and object views, capability sets of resources and organisational cells / units. Elements are components of constructs having special structure (syntax and semantics) such as dedicated languages or composite properties (e.g. behavioural rules, functional operation, information, resource and organisation) [90].

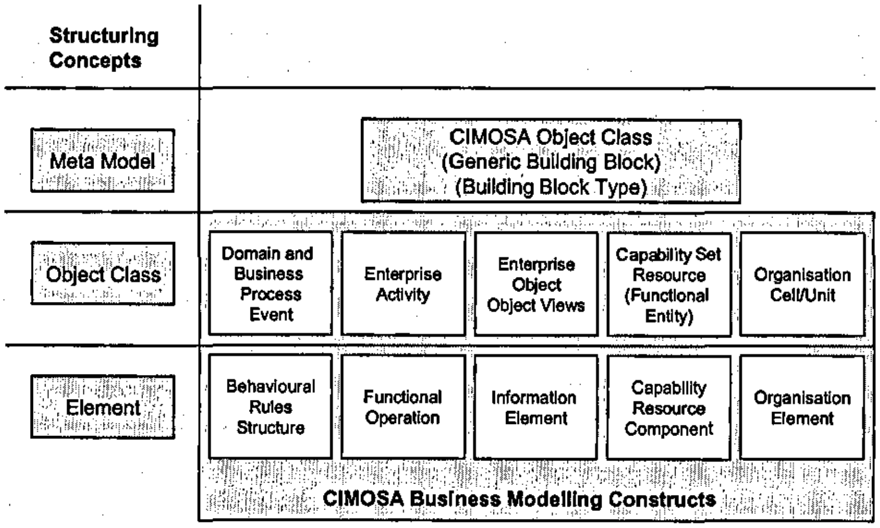


Figure 2.13. CIMOSA modelling constructs (adapted from CIMOSA [117])

IEM Approach

The IEM (Integrated Enterprise Modelling) approach uses an object-oriented approach to enterprise modelling [118]. It is based on SADT/IDEF0, using the activity box concept. IEM is based on three classes: product, resource and order. Enterprise data and business processes are assigned to objects of these classes during modelling [110]. An IEM model has two main views a function view and an information view as shown in Figure 2.14. A functional model (i.e. a business process model) is derived from the function view. It indicates the functions to be executed within the processes, the relationships between the objects and how the functions are decomposed. Tasks on objects and business processes belong to functions. The information view structures enterprise information objects covering products, orders and resources into classes and attributes to enable identification, structure, relationships with other objects, lifecycle and functionality issues to be addressed.

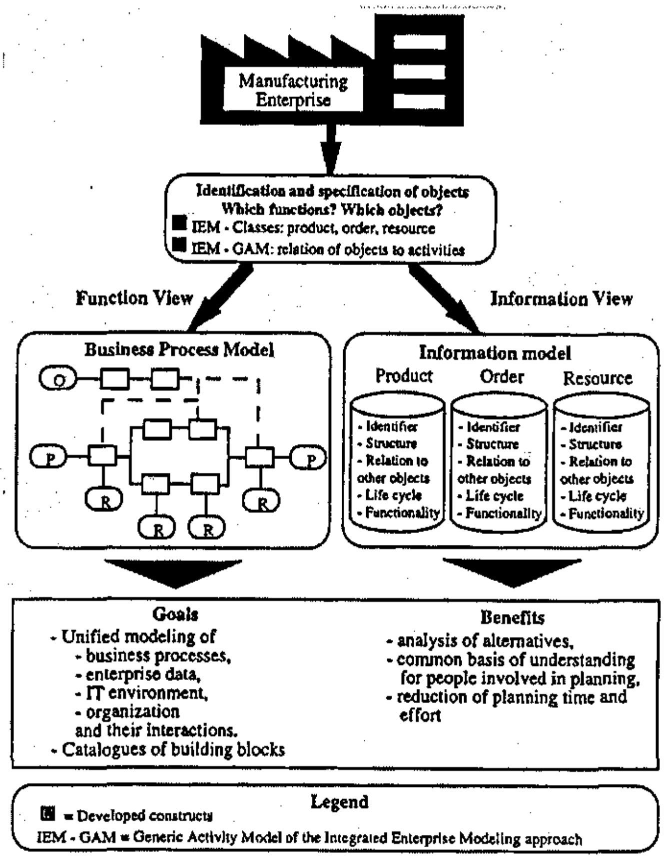


Figure 2.14 IEM views of an enterprise model (Source: Vernadat 1996, [90])

2.6 Summary

Four domains that are relevant to the research described in this thesis (illustrated graphically in Figure 2.15) have been reviewed in this Chapter. These domains are: i) the requirements of the automotive industry, ii) the Component Based Approach (CBA) targeted as a solution for the automotive industry, iii) evaluation methods and iv) enterprise modelling in terms of formal representation and dynamic modelling.

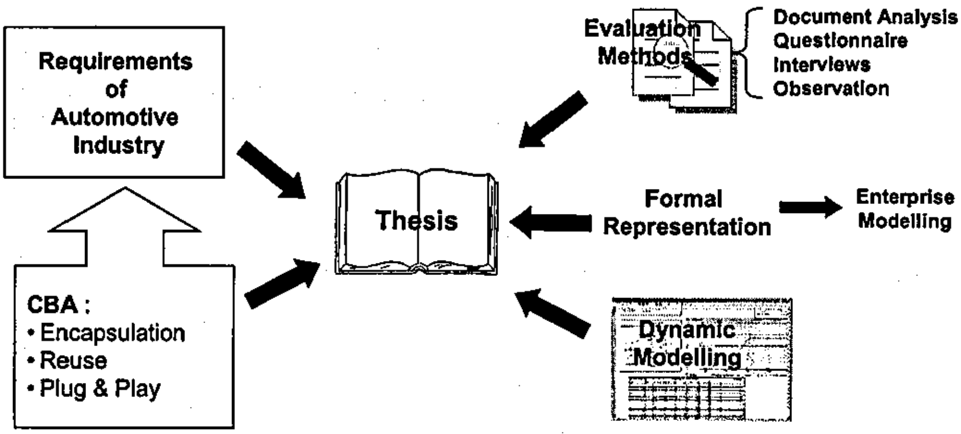


Figure 2.15 Summary of the issues of concerns address in this thesis

Developments in the automotive sector indicate that the vehicle manufacturers (VMs) are facing challenges from i) globalisation, ii) a shorter product development time, iii) demand for product variation and iv) cost pressures. These pressures are passed on to the VM's suppliers when building the machines for producing new engines. A closer examination of the process of machine design and build shows that the suppliers have overcome several problems. These problems arise mainly from a lack of effective and efficient tools to support product development.

A set of manufacturing requirements (e.g. the need for reusable technology) for the process of machine design and build has been established based upon perceived opportunities for process enhancement and improvement. Arguably of greater importance is the additional requirement to understand the impact of new processes or technology on current operations. It is the author's belief that such an evaluation requires a formal way of representing systems including a reference architecture, a modelling method and a dynamic modelling capability.

The experience of the software industry in the development of the CBA has proven its workability and associated advantages. The CBA has several characteristics, namely, encapsulation, interfaces, reuse and plug-and-play and these characteristics are the basis for the advantages offered by CBA, namely: i) improved productivity, ii) improved time to market, iii) reduced lifecycle cost and risk mitigation, iv) assured quality and improve reliability, and v) adaptable and flexible infrastructure. There are several enablers or factors which have driven the success of CBA in software industry such as: i) availability of standard technology, ii) commercially available components, and iii) the emergence of standards and best practices. The impact of CBA development has also been discussed from the product, process, business and people perspectives. From this review of the CBA development it has been demonstrated that the advantages offered by CBA are indeed the attributes sought after by the automotive manufacturing domain. It follows that the impact of the CBA in the software domain provides a good reference for the advantages that the development of the CBA in the automotive sector should provide.

Evaluation is a common term and this thesis has attempted to provide background information on how an evaluation can be carried out effectively. The evaluation process involves four stages. The first stage, planning the evaluation, is to identify the purposes of evaluation. This is followed by a determination of the evaluation design, i.e. whether or not to carry out the evaluation as a form of experiments, case studies or surveys. The third stage is the collection of data and several common data collection methods have been discussed. The final stage of the evaluation is concerned with the analysis and presentation of the collected data.

This chapter has also presented the background required (from a modelling perspective) to enable a fuller appreciation and understanding of the impact of the CBA in the automotive domain, particularly by senior business managers and engineers. The discussion has been focussed on the foundations of enterprise modelling in terms of reference architectures and modelling methods. Enterprise modelling is an attempt to provide a valid representation of the various aspects of an enterprise in order to promote a better understanding of the enterprise. Process modelling, as part of the enterprise modelling, is a representation of the activities within an enterprise, describing the functionalities and the behaviour of the enterprise. Through the use of dynamic modelling of enterprise business processes, it is possible to capture the dynamic behaviour of the enterprise. This enables the exploration of possible changes or alternative solutions for changes in the enterprise.

Chapter 3

Research Focus and Design

3.1 Introduction

The definition and the formulisation of the research questions to be addressed in this thesis are presented in this Chapter. The discussion is focused around (i) an explanation of why the research topic deserves attention and (ii) an assessment of the research methods undertaken. The research approach that has been adopted within this thesis is also defined.

3.2 Problem Definition

In the literature review in Chapter 2, it has been shown that the automotive sector is facing increasing competition; every vehicle manufacturer or end user faces the challenge of globalisation and is constantly subjected to the pressure to reduce cost, time to market and provide increased product variation. This pressure has in turn been experienced by the suppliers to the vehicle manufacturers, in particular, the machine builders and the component suppliers. To shorten the time to market, end users are looking at ways to reduce the new product development time and consequently, a reduction in the time to develop the manufacturing machinery. The usual development time for production machinery for car engines is about 50 weeks, but the end users are demanding the machine builders to reduce this development time to 42 weeks and possibly even 40 weeks [119].

At the same time, the machine builders have noticed that market pressure has resulted in incomplete product conceptualisation and thus premature engine design and analysis on the end users' part. This has had a serious impact on the process of machine design and build (see Section 2.2.3). The machine builder faces several problems during the process (Chapter 2):

i) paper based documentation requiring translation,
ii) lack of common tool for data representation,
iii) experienced based design and implementation,
iv) changes throughout the lifecycle,

v) late verification of automation systems,

vi) no simulation or visualisation tools available,

vii) lack of remote diagnostics support, and

viii) lack of support for reuse.

Based on these problems, a set of next generation manufacturing requirements has been identified (see section 2.2.4). It has been proposed that the component based approach derived from the concepts developed in the software industry, supported by a suite of system development and visualisation tools, would be able to meet these requirements and solve the problems faced by the end users and the machine builders [32, 34, 71, 120]. However, irrespective of the capabilities of the technology, its adoption is dependent on senior managers and engineers from collaborating companies fully appreciating its impact on their businesses from functional, organisational, resource and information perspectives throughout the complete lifecycle of the design and development projects.

3.2.1 Research on Component Based Control Systems

Research has identified the application of component-based approach for automation to improve automation flexibility [71, 121-126]. In particular, Harrison et. al [32, 34] and Lee [71] have described a CBA for control system development for manufacturing machinery as a possible solution to the automotive domain issues. A vital component to the research was the development of the business case for the CBA. The work was carried out within three contemporary projects funded by the United Kingdom's Engineering and Physical Sciences Research Council (EPSRC). The industrial collaborators in the projects included Ford Motor Company (i.e. end user), Johann. A. Krause Germany, Lamb Technicon UK, Cross Hüller England (i.e. machine builders), Parker Hannifin and Bosch Rexroth (i.e. component suppliers). The collective aim of the three projects was to specify, develop, test and determine the business case for a new generation of "change capable" [34] engine assembly production systems.

The objective of the first of these projects (COMPAG project (COMponent-based Paradigm for AGile automation)[5] was to produce a practical and effective set of machine components that could be deployed in engine assembly production systems

[5] The EPSRC grant reference numbers for the project is GR/M43586

(i.e. transfer lines and assembly lines) and replace the existing programmable logic controllers (PLC). An architecture for production systems was developed, in which the components were characterised by encapsulation, interfaces and plug-and-play functionality[6]. This component technology enabled the associated behaviour of the components to be incorporated into machine design and build.

The second project, COMPANION[7] (i.e. COmmon Model for PArtNers in AutomatION), built upon the findings of the COMPAG project. This project developed a common model-based environment for the lifecycle engineering of manufacturing automotive engines. The COMPANION environment provided an integrated toolset[8] for the globally distributed engineering partners which allowed remote viewing and manipulation of the common model. The tools that were developed provided (i) component selection and configuration, (ii) machine process interlocking, (iii) visualisation of both the physical and logical machine systems, (iv) human machine interface (HMI) configuration, and (v) monitoring and simulation of the complete system behaviour.

The third project, GEMM[9] (Multimedia Environment for the Global Engineering of Manufacturing Machinery) project was focussed on the development of multimedia tools to enable an evaluation of increased interaction capability among the stakeholders.

For the purpose of this work, the three projects will be commonly referred to as CCG.

Besides the work performed at Loughborough University, industrial companies have also undertaken research into the adoption of CBA for control systems on manufacturing machinery. Siemens [127], one of the largest suppliers to the market, has introduced the Component Based Automation [19] concepts for industrial automation. Siemens proposed the encapsulation of their proprietary software contained in vendor-specific control hardware, into "technological modules" [71]. Each of these modules comprises of mechanical, electrical/electronic and software constituents. An engineering tool known as iMap [127] has been developed to facilitate the implementation. A European consortium, the Interface for Distributed Automation

[6] The definition of a component will be given in Chapter 4 of this thesis
[7] EPSRC Grant reference number GR/M53042
[8] The details of the integrated engineering tool and the common model will be given in Chapter 4.
[9] EPSRC Grant reference number GR/M25025

(IDA) [128], has also embarked upon new research into distributed automation, in which the control logic is an integrated part of a mechanical object and the control system technology and programme could be treated like the mechanical parts and be reused.

The major similarity between the CCG projects, Component Based Automation and IDA is that all parties acknowledge the growing demand for distributed control to improve agility, modularity and reduce laborious and error prone programming activities. However, the major focus of the two industrial players is on technology. No detailed analysis has been done on the business impact of changes, with the majority of the discussions focused on the technical advantages of the CBA. Under such circumstances, the end users and the machine builders have been "forced" to make decisions based on technology considerations alone. It is important that a more balanced view of CBA is attained and there is a need for a formal method to evaluate the effect of change and present the results in a readily understood format.

3.2.2 Evaluating the effects of change

It is acknowledged that the uptake of the new generation of CB control systems will not be immediate. Firstly, the CBA has to be technically practical and secondly, the business viability has to be justified. It has been shown that one of the most significant barriers in implementing research results is the development of commercial products which utilise the new paradigms [129]. The business of car manufacturing is high risk and investments are high, running into billions of dollars [2, 19]. It is no wonder that end users are often cautious about adopting unproven technology into expensive manufacturing systems that will have to be used and supported for typical lifetimes approaching ten years. Traditionally, the component suppliers have depended upon proprietary solutions for maintaining and increasing their market share. This would explain why they are hesitant to adopt new technology for which they do not own the intellectual property rights [129]. Moreover, in a multi-collaborator project such as automotive engine project, any changes in the process can have major impact on the other collaborators. This has already been illustrated in the automotive sector; when the end users took the lead to explore into international markets, the component suppliers soon followed and set up their own regional operation offices [21]. As the end users decided to adopt a modular approach for their car manufacturing and concentrate on their core competencies, they relied on the suppliers to supply them with the necessary modules or systems (see Section 2.2). This illustrates that any changes have to be supported by all the collaborators because it will affect their enterprise operation strategy [21].

Research has shown that in adopting new technologies, enterprises do not consider only the technological implications [130, 131]. Business factors such as financial commitment [132] (for example hardware cost, software cost, support systems, training cost), operation benefits [133] (for example, flexibility, productivity, shorter development time, quicker response to customers) and strategic consideration [132] (for example, market leadership, competitive advantage) are some of the deciding factors when it comes to the justification of new technology. Human factors such as skills, experience, knowledge, work practice and training, though not key deciding factors in affecting enterprise decisions, are vital aspects which should deserve more attention [131]. Human resources management and deployment have crucial roles in shaping the competitiveness of an enterprise [115]. Having the knowledge of how their human resources will be affected by a technological change is thus important to an enterprise when planning the management and deployment of their human resources and the adoption of the new technology.

The work reported in this thesis was part of the research carried out under the CCG projects and is focused on the *business factors* and *human factors* that are involved in the implementation of the CB control systems in the automotive sector. The research uses the CCG projects as case studies to evaluate the adoption of CBA for the control systems in production machinery. The focus of the evaluation is on assessing the current practice of the processes involved in machine design and build, and making comparisons with the new working practices and processes that would be necessary if CBA is to be adopted. As mentioned, any changes will affect the many collaborators in the engine producing project, therefore the research will look at the possible effects on the stakeholders, i.e. end user, machine builder and the component suppliers, with regards to their products, their work process, business and actual users of the system. However, the focus has been on the effect on the machine builders as they are the major player in the process of machine design and build.

3.3 Assessing the current method of evaluation

The existing evaluation design and data collection methods described in Chapter 2 provide a good background for the evaluation process / method undertaken in this work. At the initial phase of the research two data collection methods i.e. interviews and analysis of documents were utilised. The interviews were crucial in the initial understanding of the research requirements. They were conducted to elicit information from the stakeholders involved in the project. Through the interviews, large amounts of information were elicited. Documents were also obtained from the stakeholders for better understanding of their work practices. These methods provided the background understanding of the domain. However, to assess the impact of a CB changes on the production system design and build processes, there was a need for some form of experiments or tests to be conducted. From these experiments or testing, observations were made, looking into the changes in practices and work processes that would result from the adoption of the CBA.

A review of the current evaluation methods has indicated that there is a lack of a simple, standard method to describe formally processes and enterprise interactions (see sections 2.4.3). From the data collected from the stakeholders, it was found that there were a lot of information and resources exchanged at different lifecycle stages of the project development. Documentation of these data in standard textual form or simple flowcharts (as commonly used by the automotive project collaborators) would not provide a sufficiently clear picture of the processes involved. Moreover, comparison of current practice and processes would be ineffective and complicated without a formal representation of the data [104].

Any evaluation method needs to support readily a study of the effect of change. This could possibly be done through surveys using questionnaires or interviews with the stakeholders to assess the change. However, the main problem with the classical evaluation approaches is that there were no "CB products and tools" available for the stakeholders to appreciate their use and assess the implications of adopting the approach. In addition, the paradigm has been shown (see Chapter 2) to require a radical change to working practices and system design and implementation. That could not have been appreciated by automotive stakeholders prior to its implementation. The current data collection methods are good and workable when evaluating a programme or project that has already been carried out and they lack the ability when it comes to predicting or forecasting the effect.

These requirements gave rise to the need for adopting a structured approach to enable the modelling and subsequent evaluation of the CB impact. Enterprise modelling techniques offer structured approaches for data presentation (see section 2.5). They allow a variety of data to be formalised, for example in graphical representations, enabling a formal description of engineering processes (i.e. both current and future) and the partners' interactions involved in the development. This enables the functional capabilities, information exchange, resource allocation, human resources and organisation aspects, which must be considered during the design and implementation of machine production systems, to be represented. Representations of the *current* processes (i.e. AS-IS) and *future* (i.e. TO-BE) processes can be constructed to allow comparison. The advantages of using enterprise modelling approaches are: i) they provide a suitable way to represent the data in a clear and easily understood format, ii) they allow comparisons to be made easily, iii) they show the interactions between the stakeholders, and iv) they are able to represent processes formally. For the automotive machine domain requirements, any enterprise model that is developed has to be process-based and this model must subsequently be the basis for dynamic modelling [134]. Dynamic modelling would allow analysis of change to be undertaken. It would enable the implementation of CBA within the stakeholder enterprises to be modelled and analysed without the actual risk of being the first to adopt the technology within the domain.

3.4 Research Objectives

The research objectives of this work have been defined as the following:

i) To propose a method to evaluate structurally the implementation of the CBA
ii) To assess whether the CBA is beneficial to the automotive sector particularly in the domain of manufacturing machinery;
iii) To identify the business issues that are associated with the CBA implementation, including benefits and issues of concern;

iv) To identify the human issues (including benefits and concerns) that are associated with the CBA implementation;

With these definitions, the author would like to point out that, the term "automotive sector" as used in the rest of the thesis, will be limited to the scope of manufacturing machinery for engine production, not the whole industry. The technical aspects and details of implementing CBA under the CCG project are documented by the author's colleagues: Lee [71], Vera [135], Thomas [136] and Mellor [137] and is not within the scope of this work.

3.5 Research Design

Based on the discussion in section 3.3, a research strategy and the methods to be used in this work is outlined in the following [10] and illustrated schematically in Figure 3.1:

i) Review the implementation of the CBA in the software industry, which provides a relevant reference for the automotive industry; the topics of interest are the impact on business, human factors and industry wide implications (Chapter 2).

ii) Use of the CCG projects as the case study for the research into the effect of implementing CBA on the control systems in manufacturing machinery in the automotive sector (Chapter 4).

iii) Develop an evaluation strategy for the CBA, which involves the use of the current evaluation methods and the enterprise modelling methods (Chapter 4 and Chapter 5).

iv) Use of an enterprise modelling method to formalise the documentation of the processes in the machine engineering process. The models are designed to capture the activities, resources, information and interaction in the processes (Chapter 6).

v) Use of a process-oriented modelling approach to represent and formalise the flow of activities and therefore develop the means of undertaking comparisons between the current situation and that enable by the adoption of the CBA (Chapter 6).

[10] Part of the research work has been reported in a conference paper, reference [52]

vi) Conduct testing to determine the changes involved in migrating from the current approach to CBA approach, with particular interest in changes to work practices, users' experience and work processes (Chapter 7).

vii) Establish a framework for analysing the issues involved in the implementation of technology changes within enterprises based around four distinct viewpoints: product, process, business and people (Chapter 8).

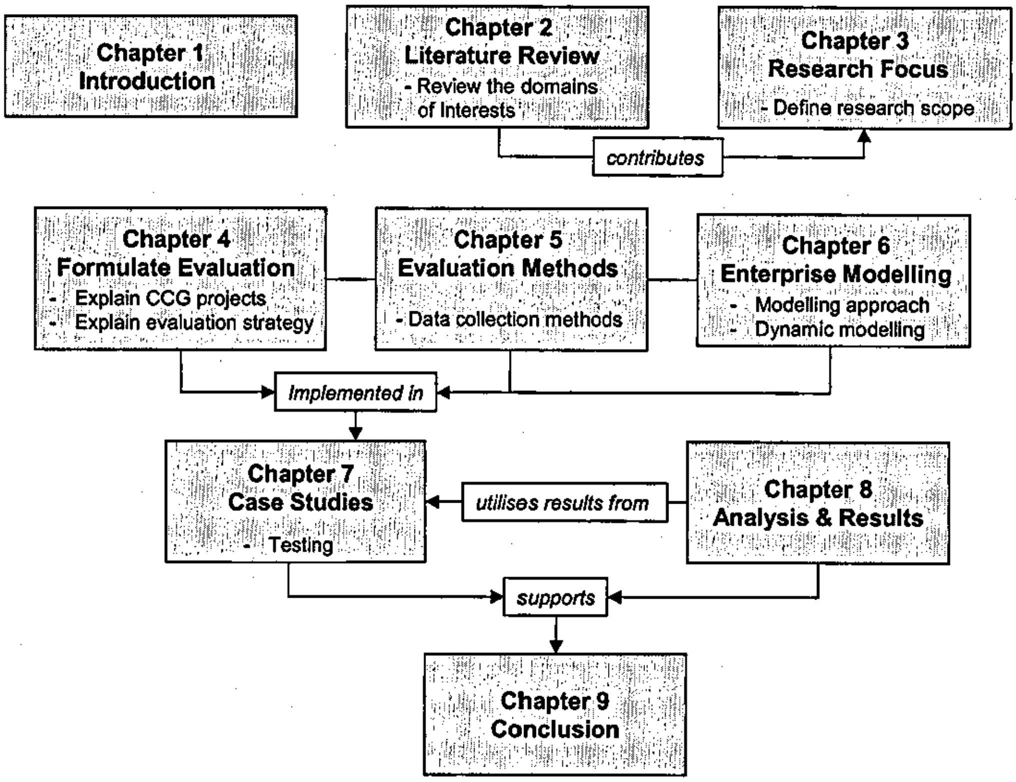


Figure 3.1 Outline of research work in this thesis

3.6 Summary

The objectives and scope of the work presented in this thesis have been discussed and specified in this chapter.

The major area of interest of this research is to find ways to evaluate the implementation of the CBA in the automotive sector and identify the possible issues that are involved with the implementation focused on the business and the human issues.

Chapter 4

Formulation of Evaluation Strategy

4.1 Introduction

Formulating an evaluation strategy and determining appropriate units and variables to be used for the purpose of analysis are vital in any enterprise analysis activity [138]. As explained in Chapter 3, this research uses the CCG projects as the case studies to evaluate the effects of adopting the CBA in the automotive sector. In the following sections the CCG case studies are presented in detail to provide a background understanding of the content and scope of the projects and facilitate the discussion and analysis. Based on the context of the case studies, an evaluation strategy, i.e. a plan to guide the overall evaluation exercise, is formulated and described. This strategy includes preparing, designing, implementing and analysing the evaluation.

4.2 CCG: COMPAG, COMPANION and GEMM

The basic units of analysis are the CCG projects undertaken by Loughborough University (see Chapter 3). The description of the CCG projects and the aims of the projects have already been outlined in Chapter 3. Within the projects, the basic concepts of the CBA were developed and an integrated engineering environment implemented to support the application development [136, 139]. Component based systems were implemented on two separate test machines located at the project collaborating partners' sites: one of them an assembly machine at Johann A. Krause in Bremen, Germany, the other a standalone transfer machine at Lamb Technicon in Mildenhall, United Kingdom. The assembly machine consists of a pick-and-place robot and a conveyor belt. The stand-alone transfer machine consists of a transfer mechanism and a machining station. The control systems on these test machines were commissioned using the CCG's CBA, based on the specifications given by the machine builders. These two implementations have been chosen as case studies to cover the complete range of different implementation platforms within automotive engine production systems e.g. assembly machines and transfer machining systems.

The areas or topics that need to be studied within the CCG case studies are:

i) the process of machine design and build;

ii) the actors and their roles in the process of machine build and design;

iii) the concerns and requirements of the component supplier, machine builder and end user in the process of machine design and build;

iv) the changes in the process associated with the implementation of the CBA;

v) the impact of these changes;

vi) the usefulness of the tools developed in the CCG projects and

vii) the effect of the capability to undertake remote diagnostics.

The detailed content of the CCG case studies that are relevant to this work are presented in the following sections. The architecture of the CB approach proposed by the project, the tools that had been developed and also the perceived benefits by the implementation are also detailed to enable understanding of the detailed evaluation results presented in Chapter 8.

4.2.1 CCG's Definition of Components

Under the CBA approach, a typical definition / specification of a component is that it is a mechatronic device, i.e. a device that is represented by both hardware and software capability. The functions and "know-how" for operation is embedded into the physical device, making it an intelligent component, having the ability to decide "what to do", "how to do" and "when to do" a task [71].

Components within the CCG projects have been designed to possess "plug-and-play" capability i.e. they can be integrated readily with other components and be used directly to compose the desired manufacturing automation system / systems [140]. A component communicates through external interfaces which include terminals (ports), services and event notifications [120].

4.2.2 Architecture

Under the CCG's CBA, the architectural composition of a system (i.e. a transfer line or an assembly machine) consists of sub-systems, modules, components and elements, as shown in Figure 4.1. A complete machine is defined as a system whereas elements are the most rudimentary units within the system. Elements describe the control behaviour, which is represented using state transition diagrams [71]. Examples of elements are the conditions for an actuator to move or for a sensor to change its state (e.g. a part present sensor state could be true (1) or false (0)). Components consist of both the physical devices (i.e. the hardware) and the software (i.e. one or more elements). In Figure 4.1 the example of the component given is a machine clamp and the control logic embedded in the clamp. Modules are the composition of several of components and are the smallest units to be associated with a geometrical model representation (i.e. a CAD based 3-D representation in VRML[11] for example see Figure 4.1, Vera 2004 [135]). One or more modules make up the sub-systems (e.g. transfer subsystem, wingbase subsystem or a fixture subsystem), which in turn are composed into a system.

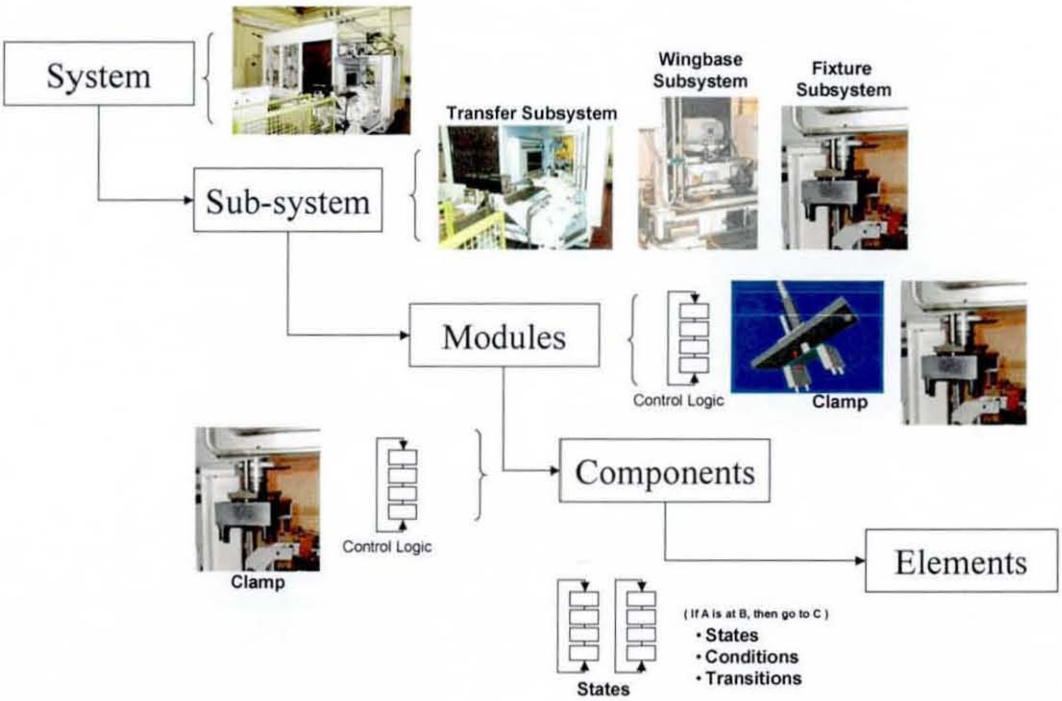


Figure 4.1 Architecture of systems with pictorial examples within the CCG projects.

Based on this architecture and the characteristics of the components, the engineering data (e.g. control data, actuators physical characteristics and performance) contained within each component and required for the implementation of the various aspects of the machine can be readily made available to all the stakeholders in the system development.

[11] Virtual Reality Modelling Language. The 3D visualisation will be discussed in the later section 4.2.3.

4.2.3 Integrated Engineering Tools for the CBA

An integrated engineering tool has been created to facilitate the implementation of systems based on the CBA. This engineering tool consists of four environments: i) control implementation [71], (ii) design and engineering [136], (iii) visualisation [141], and (iv) human machine interface (HMI) [137]. Underlying these four environments is a common database which serves as the repository for the components created and configured within the different environments [33]. Figure 4.2 shows the different environments and the common database which contain all of the information required for the formation of a common data representation model. Each environment uses the same component data stored in the common database. The common database ensures the system data are consistent within the different environments. There is no direct coupling between the four environments and each of them is independent of the target system. The tools in each environment have been developed independently. The four environments form an integrated environment, which supports the design, modelling and implementation of the CBA.

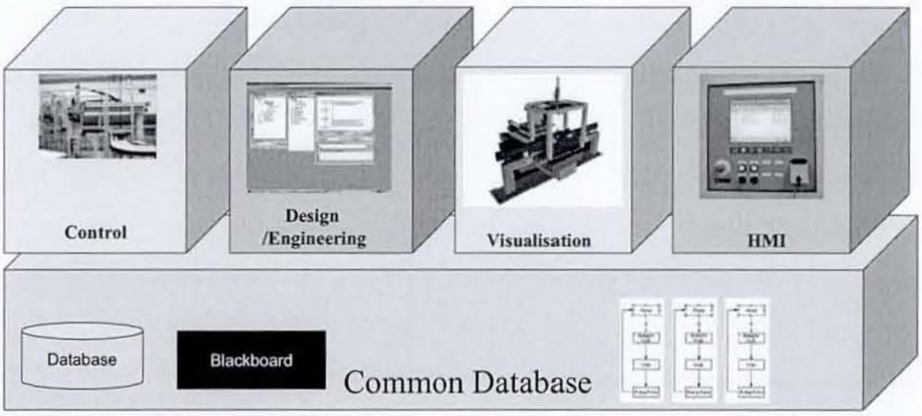


Figure 4.2 The different environments developed within CCG: control (runtime) implementation environment, design / engineering environment, 3D visualisation environment, HMI environment which supports the formation of a common model.

Control Environment

The control implementation environment (i.e. runtime environment) is the environment used for executing the control logic to drive a machine. A distributed control architecture has been adopted in this environment. The components are networked in a distributed architecture and can communicate with one another in a peer-to-peer fashion

[140]. The system does not require a master or central controller to coordinate the manufacturing operation; each component knows what (internal state machine), how (control/actuation knowledge) and when (internal and external component states) to perform the required task [140].

Design and Engineering Environment

The design and engineering environment provides the support for the lifecycle of machine design and build. This environment is used to enable the definition and configuration of a machine, simulation and the monitoring of machine activities.

Figure 4.3 shows an example of the engineering environment developed in CCG, referred to as the Process Definition Environment (PDE) since the environment enables the basic components to be configured to support the defined engineering process. A machine is created by defining the sub-systems that comprise the machine, using the machine tree hierarchy window. The machine parts list could be created either using the components catalogue or the modules catalogue. Once the part lists had been built up, the behaviour of the machine is specified. The behaviour of an individual component within the application is described within a state transition diagram (STD) [142]. The STD represents the states of an element, i.e. either a two or four (or more) state element. A four states actuator is illustrated in Figure 4.3 in which the actuator cycle moves through retracted, moving component, extended and retracting. The conditions for the element to change its state are termed the sequence interlocks. Logical combinations (i.e. AND / OR) of sequence interlocks are used to configure the desired machine application behaviour. For example in Figure 4.3, the actuator moves from retracted to moving if: i) the component feeder's component sensor shows no part present, ii) the component feeder's bin sensor shows a part present, and iii) the component transfer's transfer arm is at the left position. A component described in the engineering environment will be mapped *directly* onto an equivalent component within the control implementation environment. Although the machine behaviour can be configured by specifying the interlocks of individual elements as described above, equivalent behaviour can also be defined using a timing diagram representation as shown in Figure 4.4. The timing diagram shows the sequence of progression within the application logic similar to a Gantt chart representation for project activities [32]. The transitional state is represented by a rectangular bar (the length of the bar indicating the time taken to perform the state transition) and the conditions are represented by diamonds or oblongs. The overall sequence of the timing diagram flows from left to right and from the top to bottom, providing details of the cycle time and overall time period for the application.

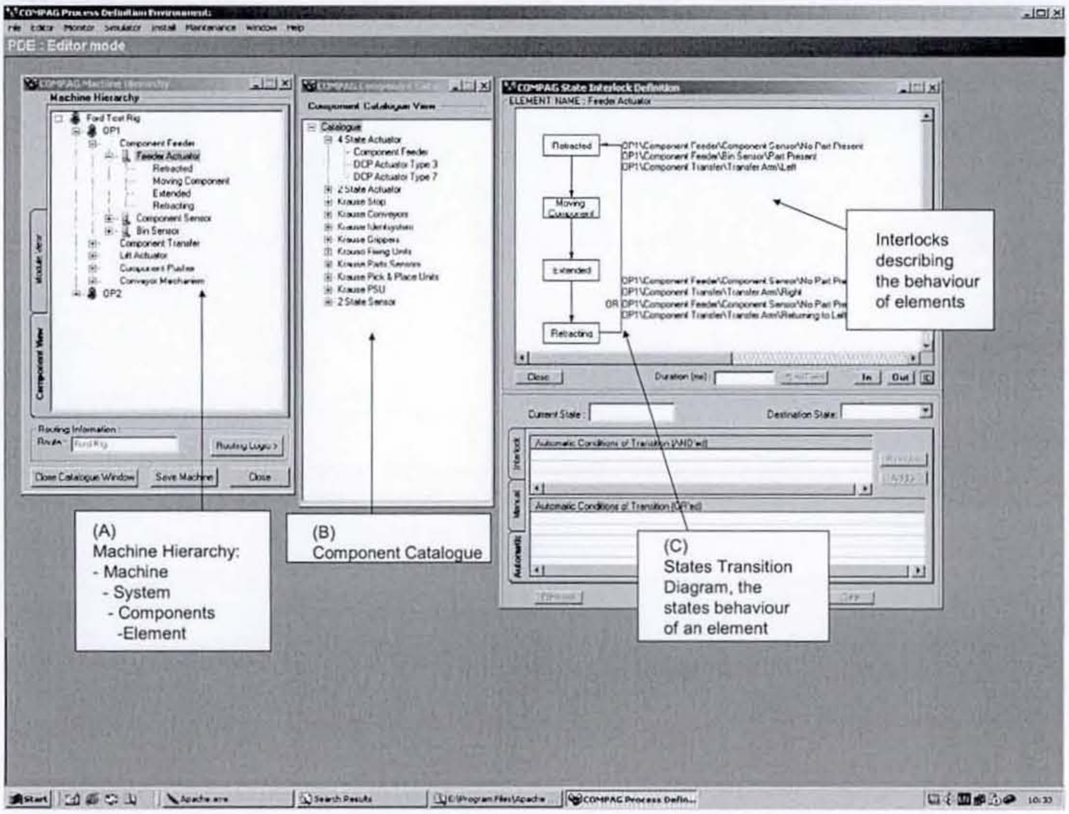


Figure 4.3 The design and engineering environment: The Process Definition Environment (PDE), which provides design, analysis and monitoring support at the various stages of machine design and build.

Using the PDE, analysis and simulation activities can be carried out before the application is installed onto the "real" control network environment. This enables error checking of the logic application early within the lifecycle. Figure 4.5 shows a simulation application within the PDE. The 3D VRML model is used for visualising the effects of the control logic on the "real" system and assessing the sequence of operation by observing the time sequence of the "physical" component actuation. At the same time, the state transition diagram can be used for checking the conditions for state changes and the timing diagram is used to monitor the sequential progression. By this process tests can be undertaken on (i) each module, (ii) each sub-system (iii) integrated sub-systems and (iv) the complete system prior to installation.

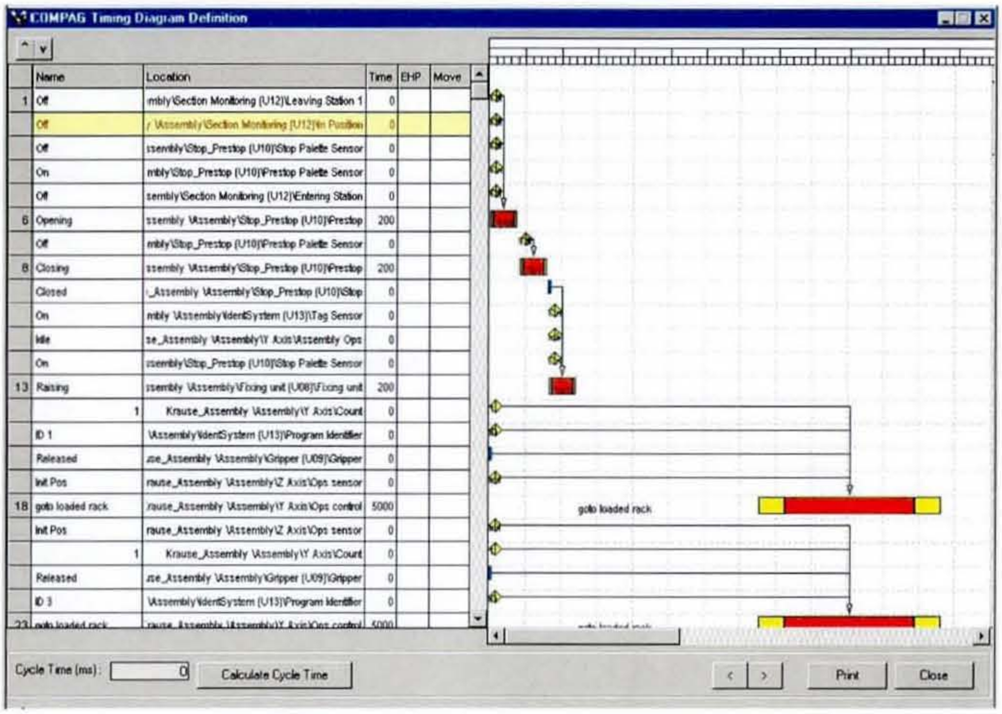


Figure 4.4 Timing diagram generated in the PDE

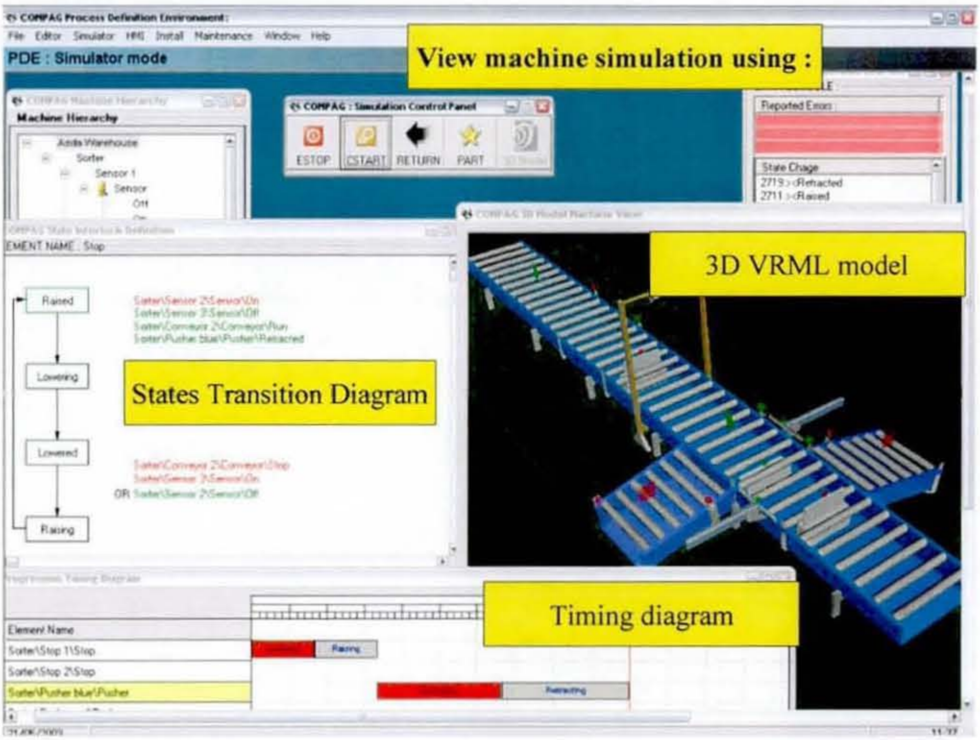


Figure 4.5 Machine simulation in PDE; the simulation of machine logic can be viewed using 3D VRML model, states transition diagram or the timing diagram.

The PDE contains the following software toolsets:

i) a component configuration tool – enables the component manufacturers to configure their components
ii) a machine configuration tool – facilitates the configuration of machine control logic, defining the interlocks and the parameters.
iii) a simulation and debugging tool – enables the virtual testing of the control logic before actual installation
iv) a machine installation tool – facilitates the downloading of the control logic onto the physical machine
v) a HMI configuration tool – provides the interface for configuring the HMI templates required by the users (This will be described in the later section HMI environment).

Visualisation Environment

The visualisation environment [141] enables a visual description of the physical behaviour of machine application to be viewed. A 3D virtual representation of the machine is created using the Virtual Reality Modelling Language (VRML) [143]. These models enable the animation of the machine behaviour that can be viewed remotely via any machine capable of supporting web browser software [141]. An example of the 3D machine representation of a simple conveyor line at Krause GmbH is shown in Figure 4.6.

The structure of the models generated in this environment is directly derived from the CB machine architecture i.e. they have the same structure as a real machine implemented using the CBA. The model can be used throughout the lifecycle of machine design and build process. For example the 3D model can be used: (i) for initial prototyping before any physical machine or control system exists, (ii) during the design phase for analysis and testing, (iii) for the training of machine operators before the physical machine is completed, and (iv) after machine installation for remote monitoring/debugging.

The visualisation environment can be run as a separate application within PDE or as a standalone software application. As mentioned earlier, a machine can be created using either the components catalogue or modules (i.e. group of components including a geometrical representation) catalogue. Thus, configuring the machine using modules would mean building a virtual machine via the 3D representations. By using the module catalogue, 3D machines are specified in a way similar to that when using the components catalogue.

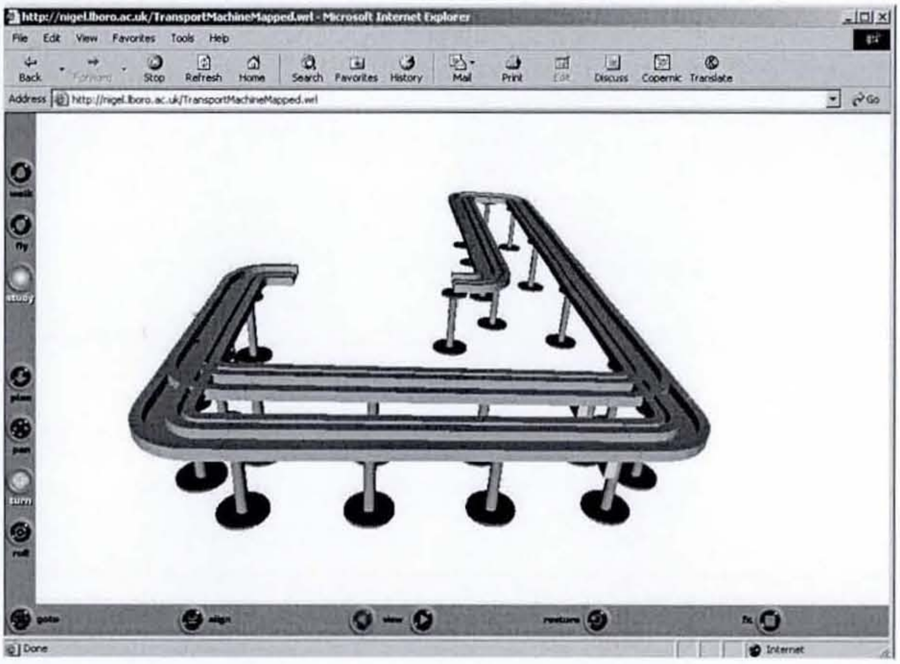


Figure 4.6 A virtual representation of a machine subsystem created using VRML, which enables the animation of machine behaviour.

Human Machine Interface Environment

The Human Machine Interface (HMI) environment is essentially the operator interface that is used for monitoring, diagnostics and controlling the operation of the virtual or real machines. The HMI environment is based on Web technologies and enables access to the CBA system via browser-based application interfaces. Such an approach enables the users to access the HMI regardless of geographical location, so long as Web technologies are available.

Figure 4.7 shows an example of the HMI created within the CCG projects based on CBA. The basic layout of the screens are determined and designed using any web-page editor. These web pages form the templates, which once generated, can be used for all of the screens that comprise the HMI. Templates enable the generated HMIs to be consistent by ensuring a common look and feel [137]. The template, when populated with the components that make up the machine, provides the required screens for machine control. This is done by using a configuration tool in the PDE, as shown in Figure 4.8.

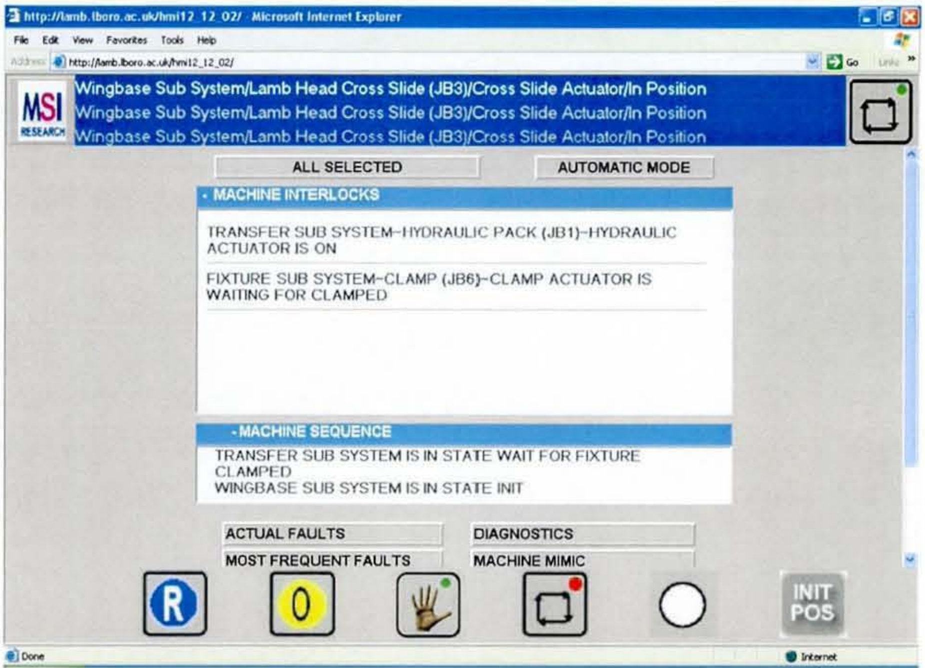


Figure 4.7 An example of the HMI created within COMPAG

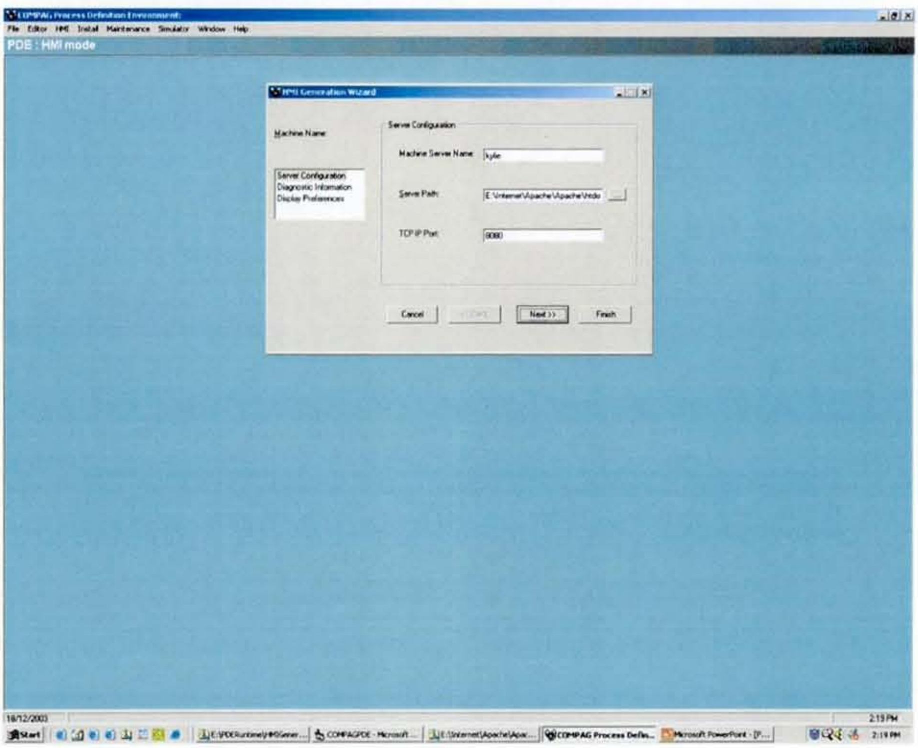


Figure 4.8 HMI configuration tool in the PDE

Common Database

Central to the four applications environments is a common database, which enables the building of a common data representation, otherwise referred to as a common model [139]. Figure 4.9 shows graphically the common database, which underpins the four different environments. The common database holds all of the information concerning the components and machine application creation (i.e. elements, components, module catalogues). When a component or machine logic is generated, it is stored in the database and becomes accessible to all of the other environments. The data represented in the components will also be stored in the common database. In addition, the HMI templates are stored in the database. Using the configuration tool in the PDE, the templates are populated with components. The implementation of a "blackboard" broadcaster allows the information from the common database to be propagated via the control network interface to the real machine and also shared amongst the different environments and other applications (e.g. Microsoft Excel, Microsoft Project).

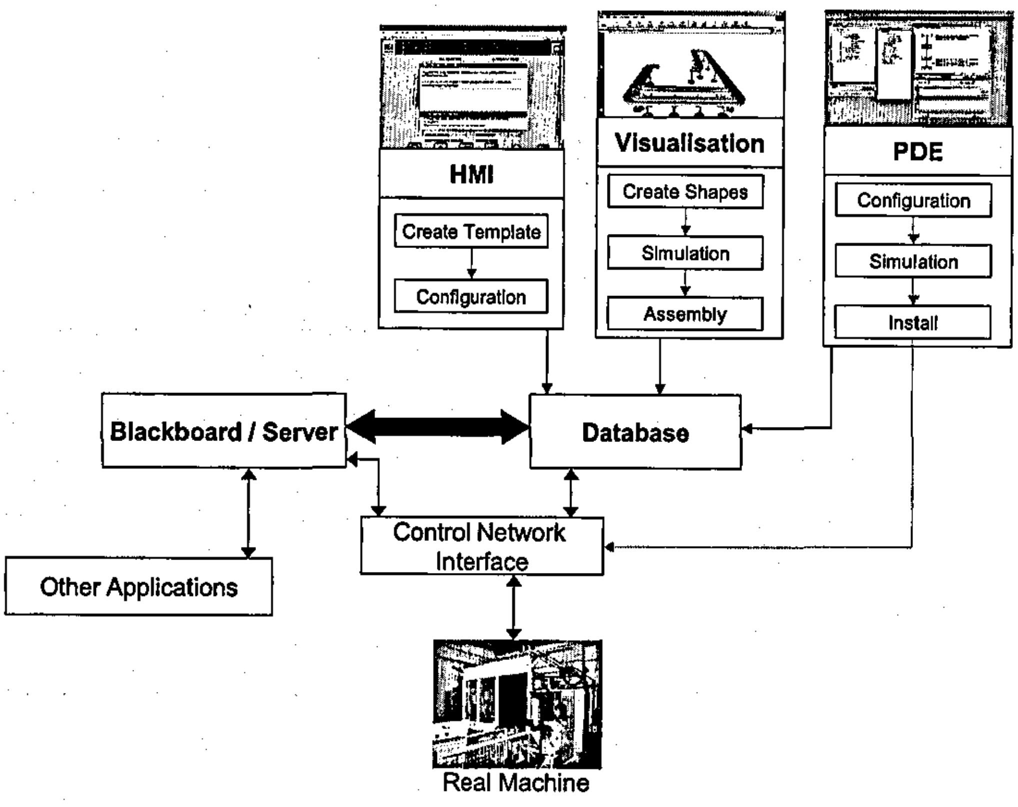


Figure 4.9 A schematic representation of the common database underlying the HMI, visualisation, design and engineering and control environments.

Common Data Representation – Common Model

The model formed using the common system data will inherently be focused around a consistent and coherent data representation. As all of the components in the machine are already defined, the common model can be considered as a logical representation of the real system. Because of the way the components are built and the architecture of the CBA, there are no losses of design semantics from design to implementation (i.e. there is no requirement for the translation of data from one form to another). Data consistency and the process of information sharing are possible throughout the lifecycle of the process. The common data representation of the system can be created at the early stages of machine design and progressively edited at later stages [33]. Through the use of Web technologies, project partners who are distributed geographically can access this common data representation to reason about and visualise alternative machine designs from their perspectives. In this way, changes on the machine design will be visible to all as and when they occur [144]. This will help to improve the communication among the project partners. At the same time, this common data representation will ensure a consistent representation of the system is used not only internally within an organisation, but also when working with external organisations [144].

4.2.4 Potential Benefits from CCG

Based on the literature review of the CCG projects, the potential benefits from the implementation of the CBA are as follows [33, 34, 134] :

i) Reusability, the basic control software is pre-written and embedded into each component and remains with it throughout its life. The components are reused by reconfiguring external parameters via the interface.

ii) Configurability: flexible links (control network variables [140]) between control elements enable the ready re-configuration of the elements.

iii) Modularity: The CBA allows the decomposition of control functionality to be matched to the required physical modularity of a machine [33].

iv) Possibility of parallel design processes by various engineering groups: the different design processes (such as control design, electrical design and mechanical design) can start in parallel instead of following a sequential order of processes.

v) Simulation of the control system before finalising design.

vi) Creation of a common model that supports globally distributed engineering partners: this will improve the interactions and communication amongst the partners by improving visibility on changes and by having a consistent representation of the system.

vii) Visualisation: the 3D VRML machine model provides a visualisation aid to the engineers during the process of machine design and could also be used for machine monitoring and debugging after the machine has been installed and in production.

4.3 Evaluation Process

The units of analysis have been described in the previous sections, providing a background understanding of the content and scope required for evaluation. The basic principle of evaluation has already been discussed in Chapter 2. A description of how the formulation of an evaluation strategy based on the context of the CCG projects was undertaken is presented in the following sections.

4.3.1 Forming an Evaluation Strategy

To conduct an evaluation involves making a plan, i.e. an *evaluation strategy*, to carry out the activities necessary for the assessment. The evaluation strategy sets the scope of the evaluation and how it is to be implemented, i.e. it is the highest level of the evaluation process. There are four stages to establishing an evaluation strategy: i) planning the evaluation, ii) evaluation design, iii) data collection methods and iv) analysis and reporting of data. Each of the stages within the evaluation strategy will be discussed and explained in the following sections within the context of the evaluation undertaken on the CCG case studies in this thesis.

4.3.2 Planning

The planning stage identifies the main issues and questions to be addressed in the study and develops appropriate methods for gathering data. It is also necessary to decide what sort of information is required from the evaluation, such as whether it is quantitative data or qualitative data, subjective or objective data (see section 2.4.5).

Planning is structured around three questions: i) what are the objectives of the evaluation, ii) who are the stakeholders, and iii) how will the evaluation findings be used.

One of the starting points in identifying the objectives is to ask questions: what kind of information is required from the evaluation, is it financial, strategic, organisational or human related; why is this information important; how is it related to the topic of research. Another way is to look at the project which is being evaluated and ask questions such as what will be the output of the project, what will it achieve, how was it achieved, what difference will these achievements make. The answers to these questions have to be clear, specific and measurable [145]. The objectives have to be clear so that an evaluator knows what to look for; they have to be specific so that they could be translated into operational terms which would help to identify the specific things to be measured. If the goals are clear and specific, the evaluator will be able to use different evaluation design and data collection methods for measurement. For example, the main goals of the CCG projects are to identify the cost and time savings that will be associated with the implementation of the CBA, the impact of the CBA to the current working practices and changes in the work processes.

An evaluation may have many objectives. Whether they could all be achieved depends on the resources that are available. These resources include the manpower available to conduct the evaluation, the time available, expenses involved, equipments and facilities to support data collection, analysis and reporting. An important aspect of planning an evaluation is to break down the timeline, tasks, expenses, personnel and materials associated with the completion of the evaluation. This enables the evaluator to make a realistic planning and scheduling of the evaluation activities.

The importance of stakeholders in an evaluation has been discussed in section 2.4.4 of this thesis. In evaluating the CCG projects, the relevant stakeholders were determined by looking at who i) uses, ii) can affect, iii) can influence the technology, and likewise, who i) are affected, and ii) are influenced by the technology. In Figure 4.10, three main groups of stakeholders have been identified from the project collaborators and grouped as the end users, the machine builders and the component suppliers. Further decomposition in the end user domain leads to the identification of the operator, plant manager, control engineer, mechanical engineer and maintenance engineer as the relevant stakeholders.

The use of the evaluation findings is very much related to the objectives of the evaluation and the stakeholders' requirements. Different stakeholders have different uses or may interpret the findings differently. For example, in the CCG projects, the end users and the machine builders were both interested in knowing how much savings could be made from implementing CBA whereas the component suppliers were concerned about the investment that had to be made in order to produce an initial set of machine components. Therefore, when planning an evaluation, it is important to determine: what would the evaluation findings be used for, what information is useful to the stakeholders, how could the stakeholders use it when specifying the objectives.

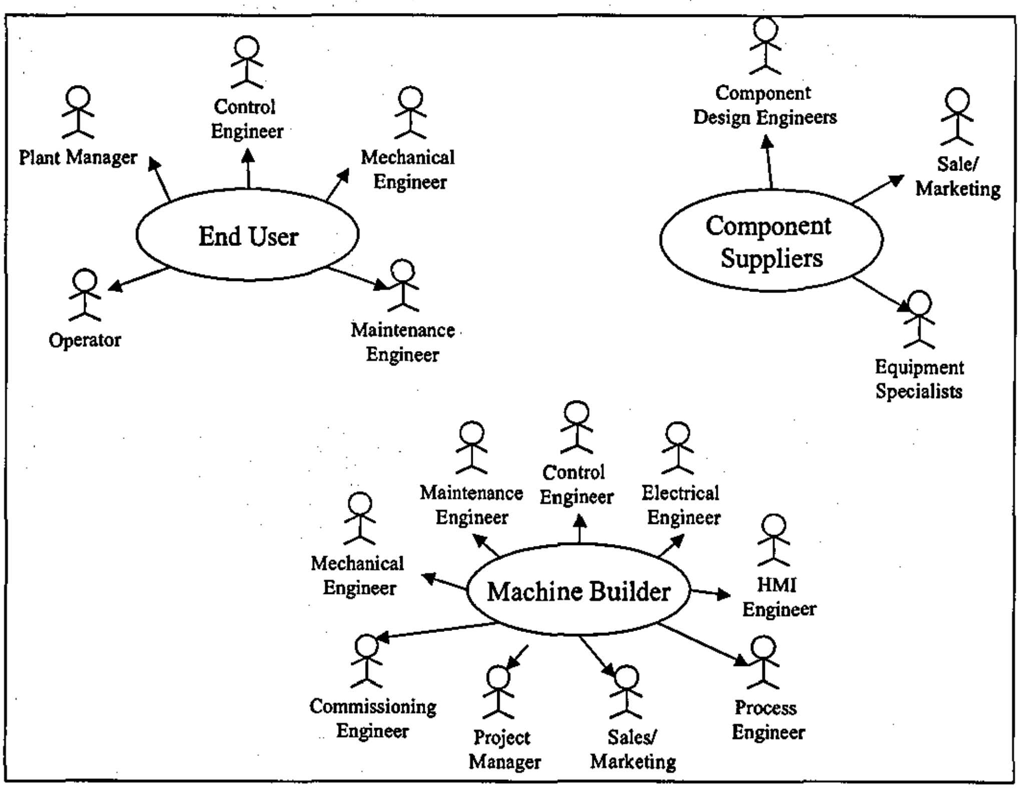


Figure 4.10 Identification of stakeholders in the CCG projects.

4.3.3 Design

The planning stage identifies the objectives of the evaluation. The next step is to identify what is to be evaluated in order to meet these objectives. This is commonly referred to as the design stage. At the design stage, the two main tasks are: i) to determine what to evaluate and ii) determine the evaluation design approach.

Having specific objectives helps to determine what to evaluate. The process is to translate the "concepts" into "operational definitions". These "operational definitions" help to specify ways of measuring the concepts [85]. For example, if the objective of an evaluation is to measure the efficiency of CBA in machine control system design, this could be done by measuring the time taken, the necessary skills and levels of experience and the number of workers that are required in using the CBA to design a machine control system as compared to using current resources and processes in automotive industry.

Evaluation designs are selected on the kind of problems to be addressed and the specific aspects of the problem under investigation [64]. Different approaches could be used in each evaluation. The important point is to select a design approach which is fit for the purpose. The decision on which evaluation design approach to use in terms of their relative strengths, weaknesses and suitability for different purposes have been discussed in Chapter 2. Different design approaches were adopted in the evaluation of the CCG projects. ***Interviews***[12] were conducted at the initial phase to obtain process knowledge, detailed experiential experience and feedback as and when required. Such interviews could only involve a limited number of stakeholders at each time. ***Documents analysis*** was also carried out for background understanding of the scope of the research topic. ***Experimental designs*** were also undertaken in the form of ***tests*** and conducted to make assessments of the CB system. To gather feedback from the stakeholders, a ***survey*** was implemented. This was carried out using ***Web-based questionnaires*** and sent out to the targeted respondents at the end of the project to obtain an overall assessment of the project. At the same time, ***industrial case studies*** involving the project collaborators were used for investigating the effect of CBA to the actual users.

[12] The selection of data collection methods will be discussed in the following section, Section 4.3.4

Different evaluation design approaches were adopted because of their relative strengths to elicit the different types of information required within the design activity (see Chapter 2). Moreover, the different approaches enabled different sets of data from different perspectives to be determined and thus allow the consistency of the data to be appreciated.

4.3.4 Data Collection

The data collection methods selected depend on the kind of data that need to be obtained and practical considerations related to time, resources and access to the sources of data [64]. Each data collection method has its concerns, strengths, weaknesses and constraints. As in the case for determining the evaluation design approach, it is a matter of selecting a method best suited for the tasks, within the limits of the resources available.

In general, when attempting to understand a topic, the first step is to read about the topic, i.e. by analysing available documentation. All research work involves to a greater or lesser extent the use and analysis of documents [64]. These documents are secondary data (see Chapter 2) which are produced by others for other purposes, which could be from journals, archives record, newspaper, magazines, work procedures.

Some methods are more likely to be associated with quantitative data, such as questionnaires. However, methods such as observation, which are likely to be associated with qualitative data, can also be used for quantifying data. An example would be to record the frequencies and durations of events such as the number and the type of design changes that occurred during a project for machine design and build. Therefore, different data collection methods could be used for different purposes depending on the objectives of the evaluation. The type of data collection methods used in this research has been described above in section 4.3.3. A simple guide to the reasons why the various method were used would be [89]:

- Documents used to obtain background information;
- Interviews and observations used to find out what people do;
- Interviews (if targeted respondents are few), questionnaires (if targeted respondents are many) used to find out what people think;

- Experiments or testing used to obtain objective quantifiable data.

4.3.5 Analysis

The final phase within the evaluation strategy is the analysis of the data. Data that has been collected has to be interpreted because data in their raw form generally do not convey much useful information. It might be necessary to consider the method for analysing the data during the stage of deciding the data collection methods. For example, to find out what people think, interviews might be used to collect qualitative and detailed data from smaller number of respondents whereas questionnaires could be used to gather information from a larger number of respondents where less detailed data is sufficient.

The raw data had to be converted into some form that can be easily studied. In most cases, data is converted into the form of documents, transcripts or some graphical representation [146]. In this research, the technique of enterprise modelling and dynamic modelling (i.e. simulation) has been adopted for representing and analysing the data (see Chapter 6). The information for constructing these models has been derived from the other data collection methods as outlined above (see Chapter 5). The enterprise modelling approach was used after the data had been organised and summarised. Enterprise modelling enabled the abstraction of the key information on, for example, the enterprise organisation, work processes, activities, physical and human resources, resources interaction and enabled their representation using graphical modelling constructs. Such models have facilitated the detailed analysis of the existing processes, i.e. AS-IS process, and the future processes, i.e. TO-BE process (see Chapter 7).

One of the advantages of using modelling methods for representing the data is that the modelling constructs can be used as the basis for dynamic modelling. Dynamic modelling used in this thesis has allowed: i) the 'visualisation' of the interactions in the processes and ii) quantitative values to be derived for the parameters that are important to the stakeholders with respect to the evaluation exercises (see Chapter 6). Different scenarios have been created and tested using the dynamic models.

4.4 Summary

The main case studies used in this thesis, (i.e. the CCG projects), have been described in this Chapter. The architecture of a machine proposed by the CCG projects, the different application environments (or the engineering tools created under the projects) and the potential benefits from the CCG projects have been explained. A description of the integrated engineering tool comprising of the different application environments, namely, the control environment, the design and engineering environment (PDE), the visualisation environment (3D VRML models), human machine interface (HMI) and the common data representation have also been detailed.

The formulation of an evaluation strategy has also been briefly explained and outlined. The evaluation strategy includes planning, evaluation design, data collection and data analysis. The data collected within the evaluation strategy have been subjected to analysis via enterprise modelling methods (see Chapter 6).

Chapter 5

Structured Methods for Evaluation

5.1 Introduction

The formulation of an evaluation strategy has been described in the previous Chapter. The methods for data collection used in this research i.e. analysis of documents, interviews, questionnaires and experimental design are described in more detail in this Chapter. Together, these methods form the core of a structured approach to evaluation. The Chapter begins with a discussion of documents analysis and knowledge elicitation techniques and concludes with the documentation of the requirements for testing in future implementations.

5.2 Analysis of Documents

The analysis of documents tends to be the first step in any research project [69] and usually begins with the identification of the domains of interest and a review of the materials relevant to the research. Other than obtaining the information from the public domain, stakeholders can also provide useful documents for reviewing.

In this research, there are two main parts to the documentation review: i) understanding the role of the stakeholders and their current working processes i.e. the AS-IS processes and ii) understanding the implications of implementing the CBA for the stakeholders, i.e. the TO-BE processes.

In trying to understand the role of the stakeholders and their working processes, requests were made to the stakeholders in order to obtain company documentation whenever possible. This documentation has included for example organisation charts, engineering process flow charts, change management requests and records of changes made throughout the development process.

The second part on understanding of implementing the CBA covered the review of the literature in the software industry which has already been discussed and presented in Chapter 2. The focus of the analysis of the software engineering documents was centred around impact of the CBA on the business and the human factors. The review has led to the compilation of a list of potential benefits that CBA could bring to the automotive industry in terms of product, process, business and people.

5.3 Interviews

Interviewing the stakeholders was the most straightforward and convenient way of getting information [85]. It allows the interviewer to response spontaneously to the answers given by the interviewee and the interviewer is able to clarify any doubts or questions that are raised during the interview.

The interviews in this research can be broadly categorised into three phases: initial phase, intermediate and final phase. At the initial phase, the topic for discussion during the interviews tended to be more general. This is because the interviewer or evaluator needed to establish a background understanding of the topic under study. This also implies that at the initial phase, the interviews were less structured. The questions asked at this phase were general, such as, understanding what is the company's main business, their customers, the workers population of the company and the organisation of the company. In some cases, the questions to be asked were sent in advance to the interviewee. The questions helped the interviewee to prepare the necessary documents or information for the interview. During the interview, the interviewer stated the purpose of the interview and the information that was sought. The order of the questions was not fixed and the interviewees were allowed to develop ideas and speak widely on the topic raised [77].

At the intermediate phase, when the interviewer had sufficient understanding of the interviewee (i.e. the company), the interviewer was be able to focus on the main topics, such as understanding the working process within a particular department, the working practice of the personnel in the department, how they deal with the customer's request. Interviews at the intermediate phase tended to be more structured, focusing on particular topic and more specific questions were asked. These one-to-one interviews were often undertaken as a result of previous discussions. The interviewees were in most cases senior engineers (i.e. mechanical, control, electrical) or the heads of the departments of interest (i.e. engineering, marketing, manufacturing, controls).

At the final phase of the interview process, much of the information required had already been obtained and the interviews tended to be shorter. The interviews were centred around clarification of information that in some cases was also obtained via phone calls and emails instead of face-to-face interviews.

It has been recognised that interviews, as a form of data collection method can be subjected to biasness and inaccuracy [85]. However, in this research, the possibility of bias is not a major issue because the main focus of most of the interviews was on the "well-defined" process of machine design and manufacture. The interviews were directed towards describing the working processes of the stakeholders. On the question of accuracy, the errors were reduced through the development of process models and obtaining regular feedback on the models from the stakeholders for verification (see Chapter 6).

5.4 Experimental Design - Testing

The issues surrounding the implementation of a system include not only the technical system i.e. hardware, software and infrastructure, but also the organisational and social (human) issues [147]. Most of the evaluation methods reported in the research literature (Chapter 2) facilitate the evaluation of the actual implementation of a system. This does not necessarily mean that the all of implications and issues can or will be highlighted early within the system lifecycle. A method which enables the researcher to be able to foresee the opportunities and problems prior to implementation (or as early in the lifecycle phase as possible) is required. In the tests used in this thesis, a task was set to assess how the CB system would be used and "user representatives" (i.e. either the user or someone who acts as the user when the actual user is not available) have been used to undertake the tests where appropriate. Future performance metrics have been determined via scenario analysis [148].

5.4.1 Scenario Analysis

Scenarios are informal methods that have been used by systems developers in designing new systems for many years [149]. They are often used to explore concepts and ideas, helping the developer to gain better understanding of the system they are developing and to communicate the performance results to others. Scenarios have also been used for training, documentation and for usability testing [150, 151]. Scenarios can be used at different stages of the development process [152]. Each scenario has its own aims and objectives and can thus be focused on different aspects of design, or analysis, implementation or test be directed towards different levels of detail [153]. There are many different methods used to represent scenarios, ranging from text based narratives, story boards, to games and role playing, video and rapid prototyping tools [154, 155]. They are created and used to represent or show how a system or application is being used [97].

Carroll [149] defined scenarios as stories about people and their activities. They are stories that describe the instances of use, a user's view of what happens, how it happens and why. They are based on the work activities of prospective users [156]. Scenarios are concrete and explicit, i.e. they are abstract ideas; they are enacted scenes used by developers and evaluators to envisage the events that occur or might occur when a system is designed, implemented and operated i.e. to describe what the system will do and how it will do it.

Scenarios have various elements [148]. These elements are setting, agents or actors, goals or objective, actions and events. *Setting* is the circumstances in which an activity takes place. The humans involved in the activity are the agents or actors. Each agent or actor has *goals or objectives.* These are the changes that agent wishes to achieve in the circumstances of the setting. Every scenario involves at least one agent and one goal. When more than one agent or goal is involved, "they may be differentially prominent in the scenario"[149]. Things that the actors do, things that happen to them, changes in the circumstances of the setting are the *actions and events* in a scenario. These actions and events often will change the goal of a scenario.

To carry out testing to "envisage" how a new design system / paradigm would or could be used when put into actual use, future implementation / operational scenarios need to be created [157].

5.4.2 Future Implementation Scenarios

Future implementation scenarios provide an "insight" into what will happen when a system is implemented. They will define the system, the task, the interaction between the users and the system and the responsibilities of the stakeholders [147]. The scenarios are used to examine the usage of the system by exploring the implications in terms of changes, advantages and disadvantages for each group of stakeholders who could be affected by the system [158].

A schematic of the method used in this research is shown in Figure 5.1. The process involved the documentation of the current problems faced by the users and an outline description of the "AS-IS" technical systems. The perceived benefits of the CBA were then analysed and mapped onto the aims and functionality of the "TO-BE" technical system. At the user identification stage, the users, their role responsibilities and role relationships were identified. Usage scenarios were created based on the perceived benefits of CBA, the problems faced by the users and knowledge of how the "TO-BE" system could be used. The next stage was to interpret and assess the usage scenarios. The aim was to produce an assessment for each of the major stakeholders. The normal mode of undertaking the evaluation was to ask the actual users to evaluate the proposed system from their perspectives [157]. However, this was not always be possible for various reasons such as geographical location, lack of resource (e.g. time, cost) or political reasons. In such situations, user representatives were asked to play the role [147, 157]. It was necessary that the user representatives had sufficient knowledge of how the "AS-IS" (i.e. current system) and the "TO-BE" system (i.e. possible future system) work in order to perform the tasks specified in the scenario. In this work, the targeted users of the CB system were not available for detailed quantitative testing because of i) geographical location problems, ii) project and end user 's time and cost constraints, iii) time required on the learning curve of the CBA concepts prior to being able to participate in the test was too large for busy industrial collaborators, and iv) the complete prototype system was not robust enough for actual user trials at the time the scenarios were enacted. Hence a small sample of knowledgeable researchers from the MSI research group was deployed as the user representatives for quantitative analysis.

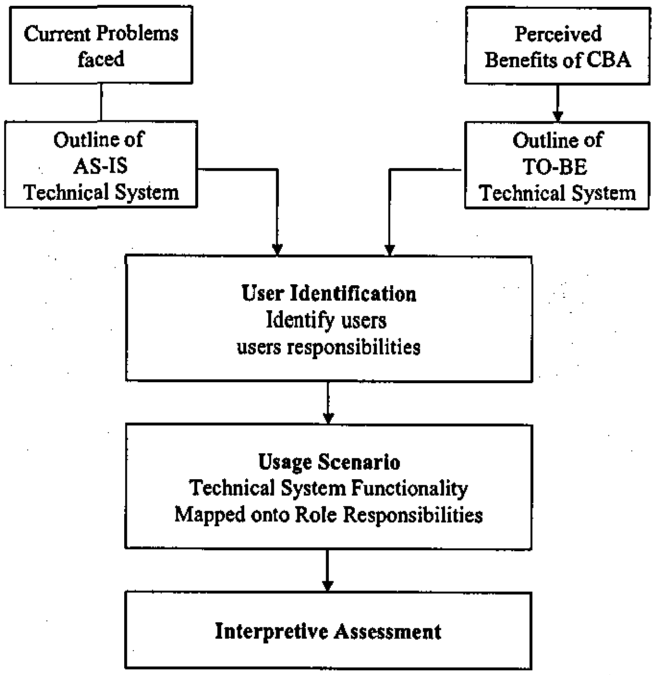


Figure 5.1 Steps taken to conduct the future implementation scenarios (Adapted from Eason 1996 [157])

5.4.3 Designing the Scenarios

The scenarios for the testing were created by mapping the perceived benefits that could be brought about by the implementation of CBA onto how the users might use the system. The objectives of the scenarios were: i) to investigate the likely implications of CBA for various stakeholders (e.g. changes in business processes, information requirements, roles and responsibilities), ii) to investigate the issues involved in users' interaction with CBA system (e.g. engineering tool usage, visualisation tool, HMI development, runtime usage) and iii) to investigate the appropriateness of scenarios in enabling likely changes / implications to be appreciated.

Figure 5.2 shows schematically i) the problems, ii) the characteristics of the CBA, and iii) the stakeholders who would have to deal with the issues associated with the adoption of a CBA for automotive machine design and build. The pressures and problems faced by the stakeholders, the characteristics and the benefits of CBA and the reasons why CBA is thought to be a solution to the problems have been discussed in both Chapter 2 and Chapter 3, with the perceived benefits of implementing CBA highlighted in Chapter 4.

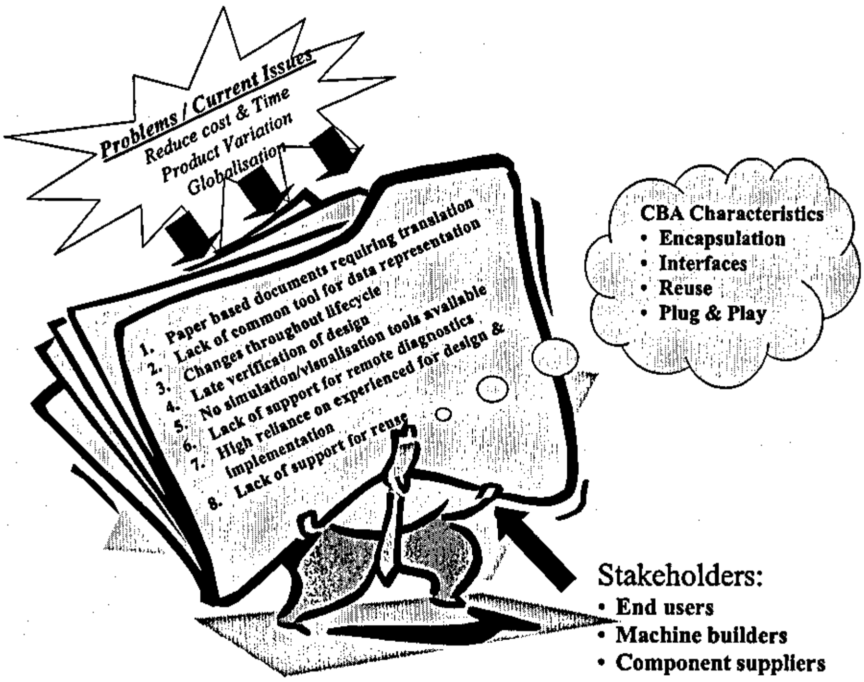


Figure 5.2 Problems faced by stakeholders and the benefits offered by CBA

Based on current problems / issues, the characteristics of the CBA and the requirements of the automotive stakeholders, several *future implementation scenarios* (FIS), representing the different uses of the system were created. (see Table 5.1). Problems that have been investigated in the scenarios are marked with a tick. The scenarios that have been created cover:

i) process translation
ii) system design
iii) modification of design
iv) system validation and analysis
v) system manipulation
vi) remote diagnostics / maintenance

Each scenario attempts to illustrate, within a particular situation (i.e. the setting), the questions asked and the task / tasks undertaken. The aim of each task is to investigate how the CBA i.e. the "TO-BE" system would be used to solve the problem / issues encountered.

Scenario 1: Process Translation

Setting: *The machine builder has the specifications to design a new machine. The engineers (e.g. mechanical, electrical and, control) involved are to design the various parts of the machine based on these specifications.*

Question: *How would the data be available and be shared amongst the engineers when the CBA has been adopted?*

The three major problems that the machine builders are currently facing are i) paper based documentation which requires re-interpretation and translation, ii) the lack of common tools for data representations and iii) a high dependence on specialised experience of a few key personnel. The consequences of these problems are: i) multiple translation of information which results in errors that are usually only discovered when the machine is commissioned, ii) machine behaviour that is not easily understood by all of the stakeholders involved in the project, and iii) the inability to reuse previous design solutions in a systematic manner. The benefits offered by the CBA is that functionality is encapsulated in systems of components, configured, analysed, simulated, monitored and maintained within an integrated environment (See section 4.2.3). This in turn enables the engineering data to be used directly by the different users. It is important to determine how much translation effort the new CB concepts and tools could eliminate.

Scenario 2: System Design

Setting: *A new machine is to be designed.*

Question: *How long will the new CBA system take to design the machine?*

One of the claims in implementing the CBA is that it will reduce the overall time taken for machine design and build. This scenario was designed to investigate the time difference that CBA will make to the overall timeline of the project. In essence, this is the major benefit of the CBA system from the stakeholders' perspectives.

Scenario 3: Modification of Design

Setting: *Assuming that a change needs to be implemented on the machine, what is the impact on the mechanical, control and electrical design and implementation processes?*
Question: *How does the implementation of CBA facilitate the changes in design?*

Changes are inevitable throughout the process of machine design and build and it is important that the impact of these changes are both understood and minimised. One of the main advantages of CBA is that the components can be easily reused. The aim of this scenario is to investigate i) whether the CBA would enable the stakeholders to react better and more rapidly to changes and ii) the issues associated with the reuse of the CBA components.

Scenario 4: System validation and analysis

Setting: *The physical machine has not been built. The engineers would like to verify the control design of the machine, reducing the burden at commissioning.*
Question: *What could the CBA implementation offer in this situation?*

The commissioning period is usually under extreme time pressures since it tends to absorb any upstream delays in specification, design and analysis. Delivery dates are rarely changed. Under the current "AS-IS" process, commissioning is the time when any inconsistencies in the system are found and rectified. One of the opportunities for the machine builders to reduce this load is by commissioning sub-systems before the final assembly and by pre-testing (e.g. virtual simulation of machine logic) of the machine behaviour. This scenario was designed to study the possibilities of having sub-assemblies tested before the final assembly and commissioning. It enabled the practicality of using virtual simulation for detecting inconsistencies in machine design to be explored.

Scenario 5: System Manipulation

Setting: *The system is processing a part and an unexpected error occurs. The on-site operator must now manually manipulate the system to bring it back to normal operation.*
Question: *What needs to be done to bring the system back to normal operation in the new CBA system?*

It is important that a CBA system is able to satisfy the functional requirements specified by the machine builders. If errors occur on the system, it is important to get it back into normal operation condition as soon as possible. This scenario was designed to look into the ease of manipulating a CBA system when an unexpected error occurs. It enabled the differences the CBA system might make to the user in terms of machine control to be assessed.

Scenario 6: Remote Diagnostics / Maintenance

Setting: *An error has occurred at the end-user's machine and the maintenance engineer needs help from the machine builder.*
Question: *How does the CBA implementation support remote diagnostics/maintenance of the machine?*

There is limited remote diagnostics/maintenance capability within the "AS-IS" process. Currently, the remote diagnostics support is phone based which often leads to difficulties in understanding and explaining the problem that has occurred on the machine. In most cases the machine builders have to send their engineers on-site regardless of where the end user production site is located. This global support is both costly and time consuming (see Chapter 2). This scenario enables the efficiency of remote diagnostics and maintenance capabilities with the CBA implementation to be assessed in terms of (i) the time and cost savings and (ii) the changes in the processes required.

The objectives, tasks and the metrics used in each scenario are summarised in Table 5.2 at the end of the Chapter. In each scenario, the effort that is required to perform the designated task has been compared between the "AS-IS" and in the "TO-BE" approaches. In the scenarios, actors have been identified (e.g. in the scenario for remote diagnostics, the actors identified are the control engineer and maintenance engineer, see Appendix E), who usually perform the specified task. During the testing of the scenarios, observer(s) were present to monitor and document the proceedings. At the end of the scenarios, feedback was obtained from the actors and the opinions were compiled for analysis [159].

The detailed descriptions of the scenarios are given in the Appendix E with the actual implementations of the scenarios detailed in the case studies in Chapter 7.

	Process Translation	Modification of design	System validation & analysis	Remote diagnostics / maintenance	System design	System manipulation
1. Paper based documentation requiring translation	✓					
2. Lack of common tool or data representation	✓			✓		✓
3. Changes throughout lifecycle		✓				
4. Late verification of automation system			✓			
5. No simulation or visualisation tools			✓			
6. Lack of support for remote diagnostics			✓	✓		
7. High reliance on experience in design and implementation	✓					✓
8. Lack of support for reuse		✓				
9. Need to shorten time		✓			✓	
10. Need to reduce cost		✓			✓	

Table 5.1 Mapping the problems to the scenarios

5.5 Questionnaire

A survey was conducted by using a Web-based questionnaire to determine the detailed opinions of the stakeholders on the CBA proposed by the CCG projects. The objectives were to find out how the stakeholders rated the CBA on the issues of: i) the practicality of CBA, ii) impact on time, iii) impact on cost and iv) their concerns with its implementation.

The main reasons for choosing a questionnaire to obtain the information was because: i) it is low cost (i.e. in this work, it was sent through email and posted on the university's website), ii) different individuals were able to complete the survey simultaneously, iii) it is fast and can be produced within a relatively short period of time, and iv) it enables standardised results to be obtained. However, the disadvantages of questionnaires also needed to be taken into account such as i) limited number of returns from an already small sample of engineers and ii) misinterpretation by the engineers.

The questionnaire was designed to be self-completed. It was set up on a website so that the respondents could access it regardless of their geographical locations. The targeted respondents were mainly the stakeholders (i.e. end-user, machine builder and the component suppliers). Attendees at the demonstration of the project results at machine builder sites were targeted. Nevertheless, these individuals were the main project manager, process, mechanical and controls engineers in the collaborators who had the most direct experience of the research and hence the most relevant input into the evaluation.

A detail and full copy of the questionnaire can be found in Appendix B. The questionnaire has a total of 9 pages and 35 questions. It was divided into two parts; the first part of the questionnaire required the respondents to provide some background information such as their organisation, their position within the organisation, the domain that their organisation belongs to. The second part was comprised of the questions regarding the CBA. This latter part was further sub-divided into six sections, as shown in Figure 5.3. Each sub-section covered a specific aspect of the project: i) common model approach, ii) the control environment, iii) engineering environment, iv) visualisation environment, v) human machine interface, and vi) overall assessment of the project. This sub-division enabled specific questions to be asked regarding the particular aspect. For example, in the section on the control environment, respondents were asked for their opinions on distributed control, embedded control and the relevance of these techniques to their organisation. Under the section on the engineering environment, questions were centred around usability, graphical programming and logic simulation. Questions on how the CBA tools might affect time, cost and skills were also included. The respondents were asked to rate their perceived impression of the impact to their organisation.

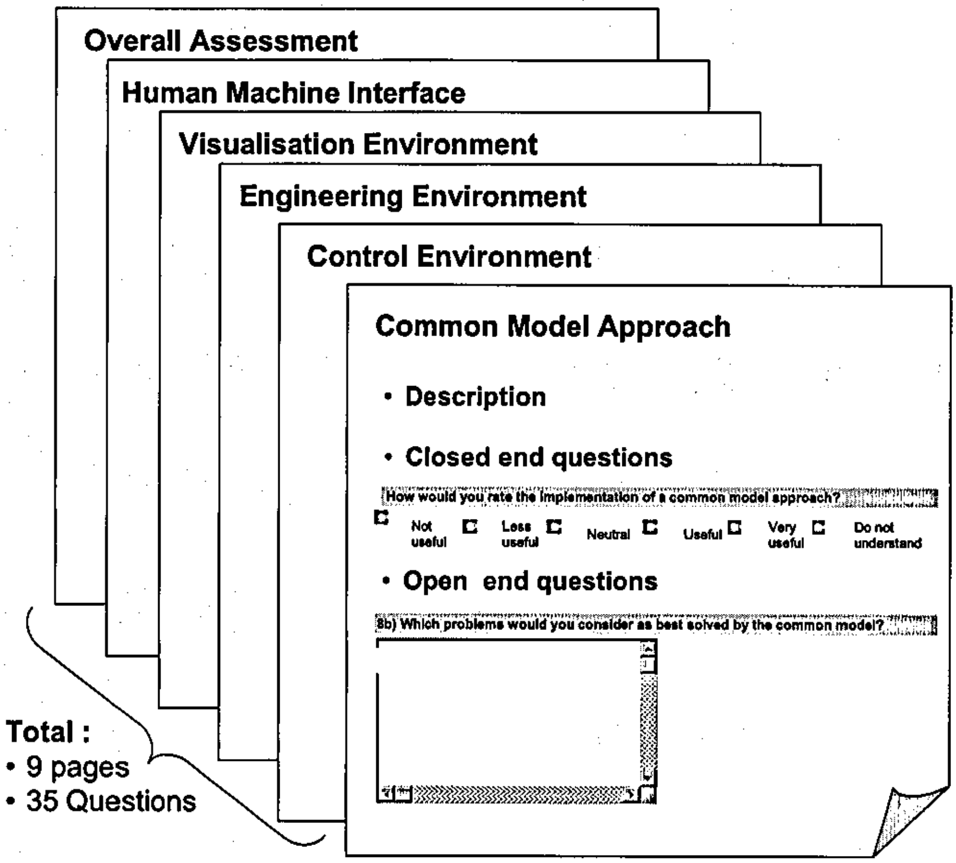


Figure 5.3 A graphical description of the different sections and the type of questions used in the questionnaire

At the beginning of the questionnaire, a brief explanation of the CBA and the research work on the approach was provided for the respondents. This provided the respondents with some background information before they proceeded to answering the questions. In each different section of the questionnaire, additional definitions and explanations were given so as to ensure a common understanding of what the terminology meant and minimised misunderstanding and misconceptions.

Majority of the questions were "closed ended" (see section 2.4.5) and "intensity questions" [82]. The main reason for using closed ended questions was to encourage a higher responding rate and secondly, to facilitate the analysis of the results. Intensity questions were used to measure the strength of the respondents' feeling or attitude towards a topic and enable quantitative information about these issues to be obtained. This is possible through the use of a rating scale, e.g. Likert scale [65], to measure the opinions. The advantage of the approach was that it provided more information about issues of concerns. For example, instead of showing that the respondents favour a particular issue, it can be shown that 5% of the respondents are strongly in favour and 70% are neutral to the issue.

Figure 5.4 illustrates an example of a typical question from the questionnaire. In this particular example, a closed ended question has been used and the respondents were asked to rate according to how useful they perceived common model approach to be. A no-opinion option, in this case using "Do not understand" was included so that respondents who do not understand the CBA can nevertheless provide opinion that can be included in the analysis.

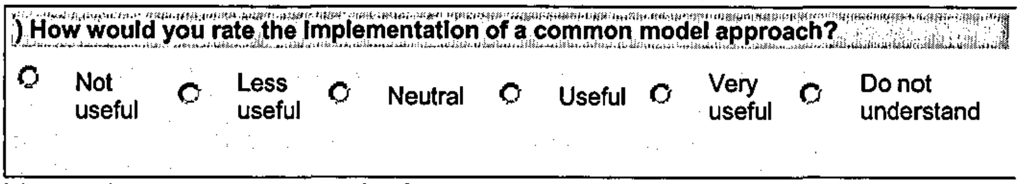


Figure 5.4 An example of a question from the questionnaire using the Likert scale

Some of the questions were "open-ended" (see Section 2.4.5). These were used when further explanation or personal opinions were required, as shown in Figure 5.5. In this example, after the respondents were asked to rate the usefulness of a common model in solving problems, they were asked to provide further information on what type of problems that they considered could be best solved by this approach. Such information is quite difficult or even impossible to acquire from the respondents if open-ended questions are not used [138].

The questionnaire was sent through e-mails and the respondents were given the Web address to the website of the questionnaire. The completed questionnaires were e-mailed to the author via the Web server. If required, reminders were sent to the respondents a month after the first email (second reminders were sent a month in some cases).

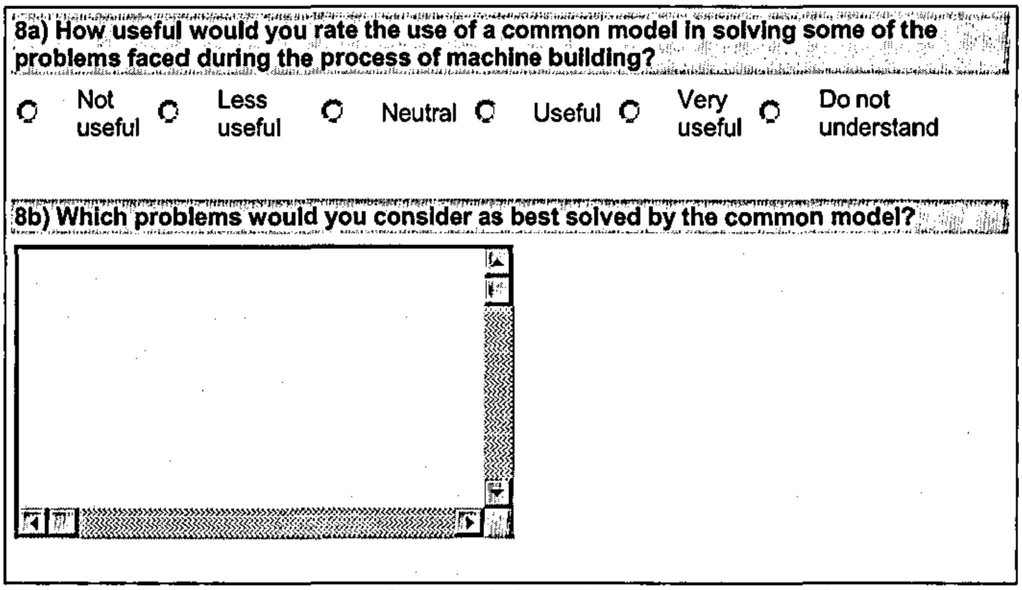


Figure 5.5. An example of the use of open ended questions in the questionnaire

5.6 Summary

The methods that had been used for the data collection in this research have been discussed in this Chapter. Several methods had been used, namely analysis of documents, interviews, questionnaires (for the purpose of the survey), and testing via scenarios analysis. This mixed method approach provides the basic structured approach for the evaluation process. The data collection and the modelling methods that were used to evaluate the data that have been employed in this work are illustrated schematically in Figure 5.6. At the initial phase of knowledge elicitation, analysis of documents and interviews were the two main methods for gathering the information from the stakeholders. The analysis of documents was supplemented by interviews with the stakeholders, who provided more detailed information and insight into the issues involved. Interviews proved to be the most direct way of gathering information. They also allowed clarification to be made with the stakeholders but were extremely time consuming and difficult to organise with the relevant stakeholders.

Two methods for assessing future implementations have been adopted in this research. One of them is scenario testing. Testing was vital to investigate how the CBA could be beneficial to the stakeholders when implemented and the results are key to enabling management buy in of the concepts and approaches required by the CBA. Scenario testing, in particular future implementation scenarios, has been identified as the method for the exploring how the users might use a new system. Future implementation scenarios, as its name implies, were targeted at studying what will happen when a system is being implemented under the CBA. The scenarios described relevant situations and questions were asked regarding what and how the new system could offer over and above the current way the scenario would have been addressed. A description of how the set of scenarios had been created for the purpose, outlining the objectives and the task in each scenario has been included. A set of metrics that are to be used in the evaluation of the test has also been identified.

The further evaluation outlined in this Chapter has been undertaken by conducting a survey in the form of a questionnaire to gather feedback from the stakeholders. The questionnaire approach was chosen because it is the easiest way to obtain this information from the global project partners involved in the research. The same set of questions was posted to all the potential respondents through e-mails. However, unlike the interviews, there was no guarantee on the response rate.

The lower part of the diagram in Figure 5.6 shows the formal knowledge representation methods that have been adopted in this thesis. These methods and the modelling approach used in this research will be discussed in greater detail in the next Chapter.

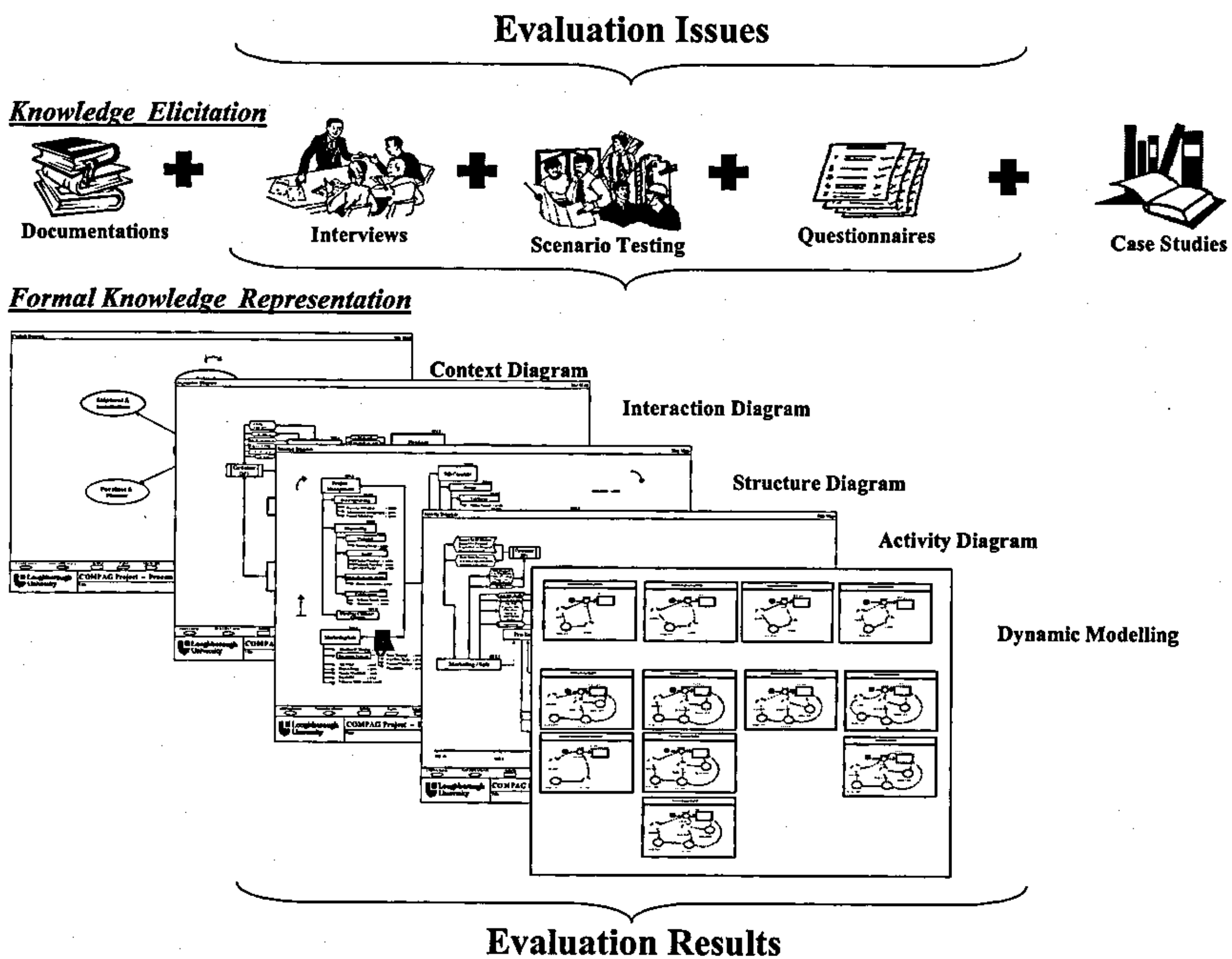


Figure 5.6 Summary of the evaluation approach adopted in this research

Scenario	Objectives	Task	Metrics
Process Translation	• To assess the level of translation that is required during the process	• To trace the process of machine design and build and count the number of times in which data are translated from one form to another	• Number of steps required
System Design	• To assess the time taken to design a new system	• Design a new system using the CBA	• Time taken
Modification of design	• To assess the complexities of changing the mechanical, electrical and control design	• To make a change to the design of the machine and observe the procedure required for making the change	• Time taken

Table 5.2 Future Implementation scenarios created, the objects, tasks and the metrics of measurements for each scenarios

Scenario	Objectives	Task	Metrics
System Validation/Analysis	i) To assess the possibilities of system verification at an earlier stage ii) To assess the possibilities of subsystem testing / incremental testing	• To test subsystems offline • To sub-assemble the systems and assess the accuracy of the control behaviours	• Time required • Number of errors
System Manipulation	i) To test the ease of manipulating the system to a desired state ii) To assess the level of knowledge required to assess (i)	• To inject errors to the machine while it is running and observe the procedures required to recover the machine	• Time required • Number of steps required
Remote Diagnostics / Maintenance	i) To assess the efficiency/effectiveness of diagnosing a machine remotely. ii) To identify suitable tools that is required for effective remote diagnostics from different users' perspectives	• Error injection and perform diagnostics using tools that enable remote control	• Time required

Table 5.2 Future implementation scenario created and the objectives and task for each scenario (continue from previous page).

Chapter 6

Data Representation using Enterprise Modelling Method

6.1 Introduction

One of the main objectives of this research is to identify an appropriate method for formally representing and analysing the knowledge elicited from industrial stakeholders. Enterprise modelling has been identified as the suitable method for the purpose (Chapter 3). The reference architecture (i.e. CIMOSA) and dynamic modelling approach used in this research are detailed in this Chapter. The architectural framework and the modelling constructs that method has been based on will be explained. A set of CIMOSA representation diagrams will be explained in detailed and illustrative use of the diagrams will also be discussed. The dynamic modelling tool (i.e. iThink™) will be introduced and details of how iThink™ constructs were used will be explained. Finally an illustration of how the enterprise modelling approach has been utilised to represent the various processes that result from the adoption of the new generation of engineering tools developed within the CCG projects will be presented.

6.2 Process Representation Model

Process representation, a sub-set of enterprise modelling architectures, has been adopted as the formal method for representing the engineering processes and the interactions between the stakeholders in this research. An appraisal of the capabilities of the range of reference architectures has already been presented in Chapter 2. A decision was made to adopt the CIMOSA enterprise engineering approach since CIMOSA has a well-established modelling architecture that has been studied, tested and validated by many academic research groups and industrial users in the manufacturing sectors [160-163]. In addition, CIMOSA enables (i) a largely top-down approach which is suitable for modelling the processes encountered in this research and (ii) the breakdown of the domain in terms of function, organisation, resources and information. Finally, the CIMOSA reference architecture has been widely studied within the MSI Research Institute in Loughborough University and this has provided valuable relevant operational experience and the development of associated methods and tools. The following section will discuss the CIMOSA framework and its modelling approach in greater detail focusing on the background relevant for the evaluation work in subsequent sections and Chapters.

6.2.1 The CIMOSA Architectural Framework

The CIMOSA Architectural Framework has been described by many authors [163-165] and is considered to be the most comprehensive of current public domain enterprise modelling approaches [90]. CIMOSA has been applied in different business domains [117].

The CIMOSA concepts are organised into three main parts [90]:

1. An enterprise Modelling FrameWork
2. An Integrating InfraStructure and
3. A Computer Integrated Manufacturing system lifecycle

CIMOSA Modelling Framework

The fundamental concepts within the CIMOSA modelling framework are usually represented by a CIMOSA cube (Figure 6.1). The framework offers the ability to describe and model an enterprise from different aspects and views. It comprises two architectures: a Reference Architecture and a Particular Architecture (see Figure 6.1).

The reference architecture provides a reference model structure while the particular architecture is a set of models that collectively represent a particular business. The reference architecture has two layers namely the Generic and Partial layers. The generic layer provides the basic constructs of an enterprise modelling language, their types, instantiation and aggregation rules. The modelling constructs of the generic layer can be used to describe models related to different life phases of an enterprise from Requirements Specification, through Design Specification to Implementation Description. The partial layer consists of libraries of partially completed enterprise models that can be instantiated and used in a particular architecture. The modelling framework also promotes the use of three key enterprise engineering principles which are derived from three dimensions (see section 2.5.3, Figure 2.7): i) the instantiation principle derived from the genericity dimension, ii) the derivation principle derived from the modelling dimension, and iii) the generation principle derived from the dimension of views.

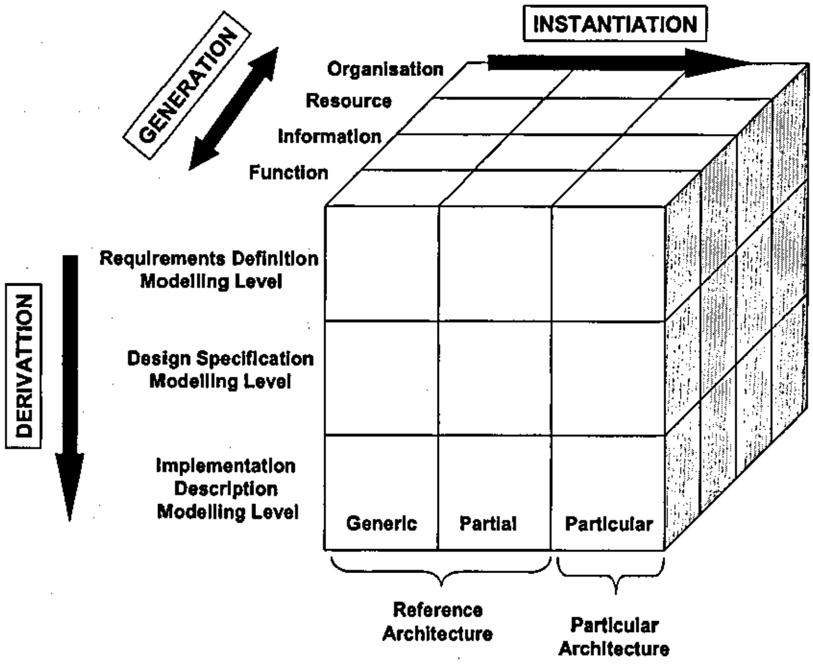


Figure 6.1 CIMOSA modelling framework (adapted from Vernadat 1996 [90])

Instantiation Principle

The instantiation principle is based on the three generic layers, which forms the Reference architecture and the Particular architecture (i.e. generic, partial and particular). The detail of each layer has been given in the previous paragraph.

Generation Principle

Under CIMOSA, for an enterprise to be fully described and understood it has to cover at least four different views: Function, Information, Resource and Organisation. This set of views may be extended if required.

Derivation principle

This set of CIMOSA principles and concepts facilitate a step-by-step derivation of models through life stages of Requirement Definition, through Design Specification to Implementation Description.

CIMOSA Integrating Infrastructure (IIS)

The IIS is a set of basic information technology services, which enable physical and application integration [109]. It provides a unifying software platform to achieve integration of heterogeneous hardware and software components of the CIM system [161]. The IIS enables interconnection of communication within the enterprise and provides support for enterprise coordination and inter-working. This set of services thus enables a CIMOSA model to be executed in a real system. However, since the IIS is mainly concerned with the integration issues concerning real enterprise systems it is not relevant within the evaluation context of this research.

CIMOSA System Life Cycle

The phases of CIMOSA System Life Cycle covers requirement definition, through system installation, test and release to system maintenance. CIMOSA considers the following phases to be the major phases of a CIM System: master plan definition phase, requirements definition, system design, system build and release, system operation, system maintenance and change, and finally, system dismantlement [90].

6.2.2 CIMOSA Modelling Constructs

The essential modelling constructs of the CIMOSA function view are events, domains, domain processes, business processes, enterprise activities and functional operations. They can be used to model enterprise functionality and behaviour [98], which are the main requirements with respect to the work outlined in this thesis.

CIMOSA enables an enterprise model to be organised into manageable modules to reduce system complexity. These modules are called **domains.** Domains interact with one another by the exchange of events (request or triggers to do something) and results (defined as views on enterprise objects and termed object views) [90]. Domains comprise a set of core processes called **domain processes** which are decided upon by the user. A domain process (DP) is an end-to-end process that can exist on its own. It is a sequence of activities of an enterprise with well-defined starting conditions and provides a measurable or quantifiable end-result [90]. Figure 6.2 shows the interaction between domains and domain processes. Once the domains have been established and their relationships made explicit, the events and domain processes of each CIMOSA domain can be defined.

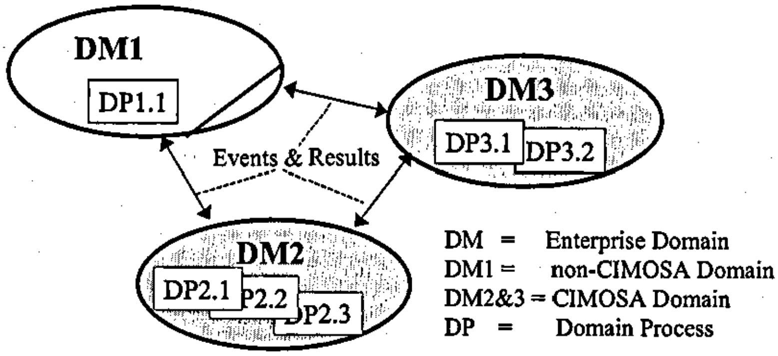


Figure 6.2 Domains with interacting Domain Process (Source: CIMOSA [11])

Domain processes can be decomposed into lower-level processes, lower-level processes into sub-processes and so on using the functional decomposition principle [109]. Domain processes are broken down into the next level of activities expressed as a set of **Business Processes (BP),** which are further broken down into **Enterprise Activities (EA)**, as shown in Figure 6.3. The decomposition is again functional based. An enterprise activity is the elementary unit of activity within an enterprise. A business process will comprise a set of related enterprise activities (see Figure 6.3).

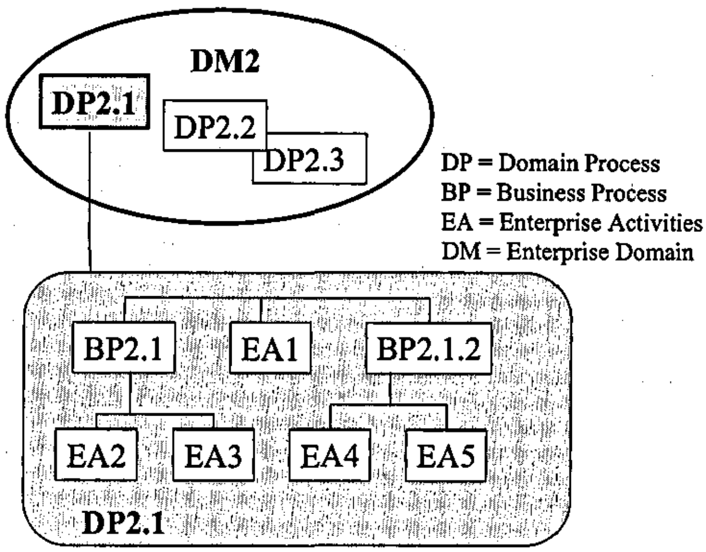


Figure 6.3 Decomposition of Domain Processes into Businesss Process and Enterprise Activities (Source: CIMOSA [11])

Domain processes are triggered by events only; they indicate the sequence of enterprise activities to be executed to realise the desired enterprise behaviour. They can be seen as a network of activities connected by behavioural rules. Business processes are similar to domain processes except that they cannot be triggered by events but must be called by a parent structure (domain process or higher-level business process) and their termination must be defined as ending statuses.

Ending statuses are values defined by users and describe the execution status of the process [90]. Behavioural rules define the flow of actions which link up the business processes and enterprise activities. They govern the sequence of execution of enterprise functionality according to the system state. They could be seen as a set of conditions and a related actions, in the form "When (conditions) Do (action)".

EAs define enterprise functionality. CIMOSA defines enterprise activities as a set of elementary actions to be considered as a whole, requiring resources and time for their full execution. An enterprise activity is defined as a set of functional operations. Functional operations represent elementary processing steps which can be executed by only one functional entity (i.e. resource).

6.2.3 Modelling Approach Adopted in this Thesis

Although CIMOSA has developed a set of constructs to describe the operation of an enterprise, it does not pay much attention to the graphical representation of constructs and does not have a firmly established notation. The diagrams used for representing the CIMOSA modelling framework in this work have been developed within the MSI Research Institute [31, 166]. The diagrams were initially developed by a number of researchers in the MSI and subsequently Monfared [31] has built on this previous work, to provide a set of four basic representational diagrams focused on the CIMOSA function view. The decomposition based representation forms the basis for the static modelling in this work. The author shall refer to this method as Monfared's Modelling Approach (MMA).

Figure 6.4 shows the basic abstraction mechanisms used in MMA. The approach contains a set of diagramming techniques to capture the activities within an enterprise. The four diagrams are namely: context diagrams, interaction diagrams, structure diagrams and activity diagrams. Collectively, the four diagrams provide complementary views of process attributes at the required level of abstraction. The diagrams could be used at any level within an enterprise i.e. they could capture the top level view and be broken down into lower level views. The activities within an enterprise are broadly classified into domains as domain processes (DP), business processes (BP) and enterprise activities (EA) (which are in conformance with CIMOSA) based on functional decomposition. There is a set of modelling notations (Figure 6.4) that has been used to represent the different processes within the enterprise, namely: activity, events, information, human resources, physical resources, finance, external links, the flow of resources or material, flow of the process and alternate flow of process.

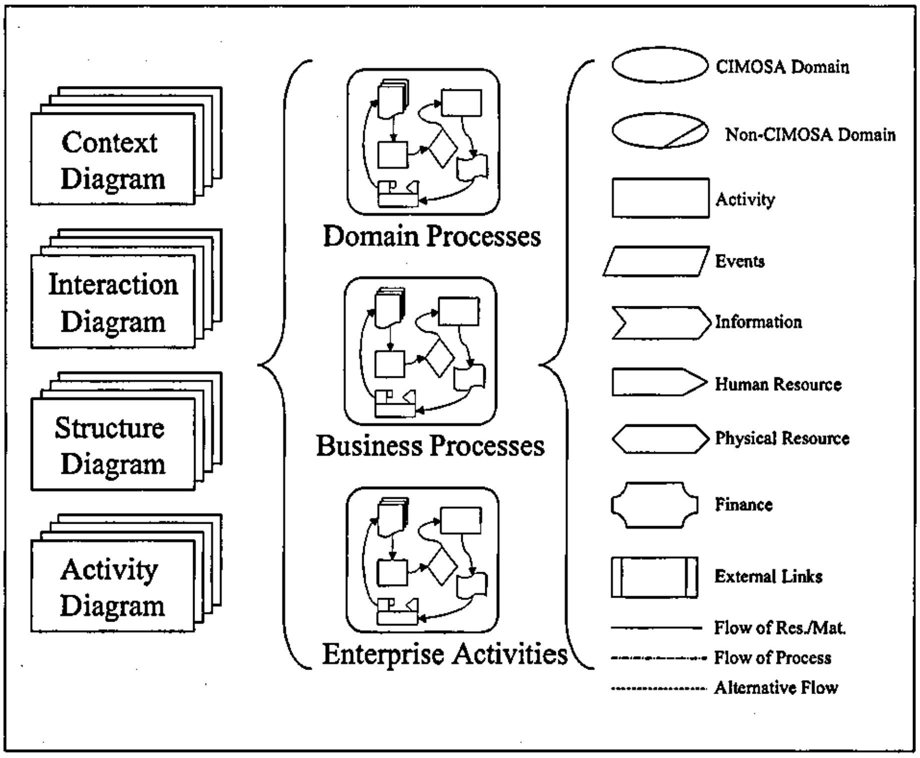


Figure 6.4 Abstraction mechanism within the Monfared approach (Source: Monfared 2002 [31])

Context Diagram

The context diagrams are organised in terms of manageable modules and define the CIMOSA domains to be modelled. The resultant modules are called *Domains*. The modules which are of concern in a project and for which models need to be created, are represented graphically as *CIMOSA-Domains*. The domains that are not modelled are referred to as *non-CIMOSA Domains*. Domains are represented by oval-shaped bubbles.

The CIMOSA domains are represented by simple bubbles, while the non-CIMOSA domains are represented by crossed-out bubbles. A context diagram can represent different levels within an enterprise. A top level context diagram can be broken down to lower level context diagram. Consequently, the domains identified in the respective context diagram can be further decomposed into sub-domains and domain processes.

Interaction Diagram

The interaction diagram describes the interaction between the domains, domain processes, business processes and enterprise activites. Interactive diagrams identify, define, organise and represent the interactions involved. Domain processes interact with each other by means of events (which typically take the form of requests or triggers to do something) and results (defined as being views on enterprise objects). The interactions between domains take the form of information exchange, human resource exchange, physical resource exchange and events.

Structure Diagram

The structure diagram decomposes each domain process into atomic functional elements of business processes and enterprise activities. These diagrams are used to identify, structure and organise business processes and enterprise activities. There is no indication of sequence of activities. Structure diagram represents the basic structure of the enterprise.

Activity Diagram

An activity diagram shows the time sequence of business processes and enterprise activities. Enterprise activities, business processes and control flows are represented by the graphical model building blocks. At the same time, like in the interaction diagram, activity diagrams also show the information exchange, human resource exchange, physical resource exchange and events but at a more detailed, local level.

6.2.4 Illustrative use MMA to Enterprise Modelling

Figure 6.5 shows a context diagram derived within this research, which consists of a domain "New Engine Project". This is a top-level context diagram of an overall project. It has many sub-domains, some of which are of modelling interest i.e. CIMOSA domains, and some which are not of modelling interest are shown as crossed out bubbles, i.e. non-CIMOSA domains. Each new engine project comprises market study, financial / technical assessment, financial arrangements, project management, engineering design, site preparation, production / assembly machines, installation and test, mass production and maintenance domains. Although the CBA can be expected to have some measure of impact in all of the domains, by far the most important within the context of this research are the Production / Assembly Machines, Maintenance and Installation and Test domains.

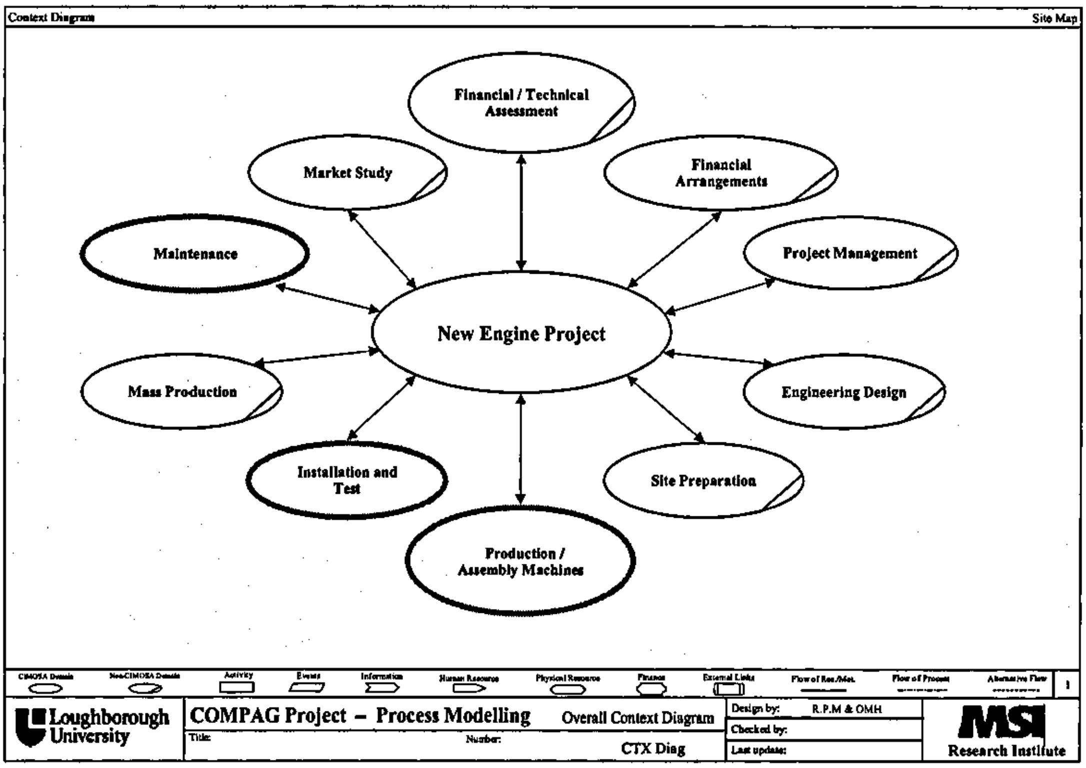


Figure 6.5 A top level Context Diagram showing the overall project (Source: Monfared 2002 [31])

The top level context diagram can be broken down into lower level context diagram(s) as shown in Figure 6.6 where the production / assembly machine domain is shown as being of importance to the end users, machine builders, component suppliers and technology vendors. As indicated in the diagram the research focus has been on end user, machine builder and component suppliers rather than technology vendor.

Domains could be decomposed to sub-domains. When domains are decomposed to the level where the core processes of concern have been identified, the resultant sub-domain is identified as the domain process. In Figure 6.6, the sub-domain "Production/Assembly Machines" is identified as the process of concern and is further decomposed into domain processes named as "End user", "Machine Builder", "Component Builders", and "Technology Vendor". A domain process can exist independently and deliver quantifiable or measurable outputs. It can be viewed as an independent unit of capability that when grouped together, results a valuable whole [105]. Each DP can be decomposed to sub-domain process, such as the domain processes for the machine builder Cross Hüller as shown in Figure 6.7. Five sub-domain processes have been identified within the DP i.e. marketing, production design, manufacturing, finance (purchasing) and sub-contracting.

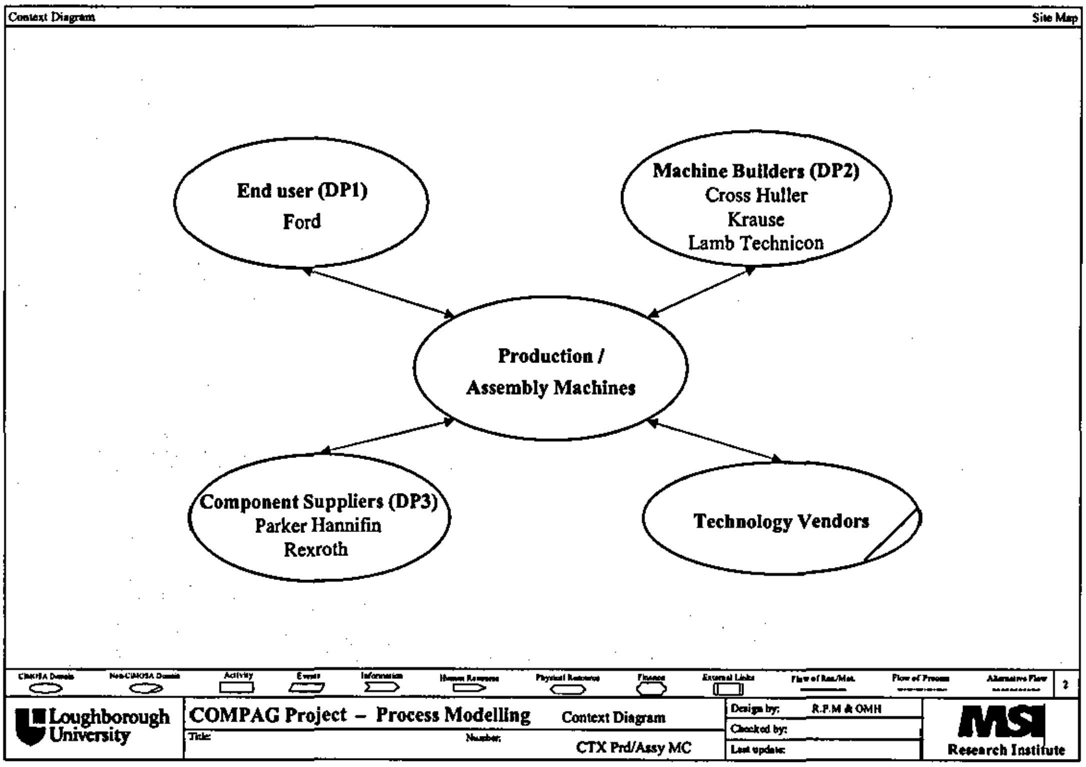


Figure 6.6 Context Diagram of Production / Assembly Machine

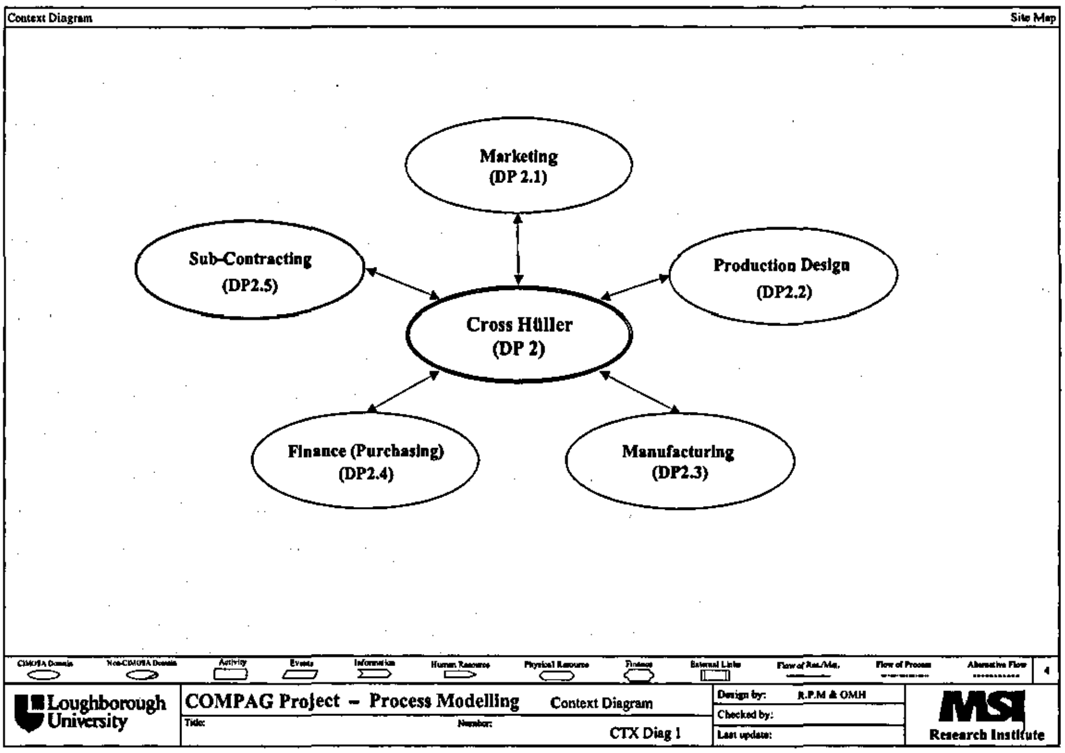


Figure 6.7 Sub-domain processes within a domain process

Domain processes influence other domains by means of events and interactions. The interaction among domain processes takes the form of information exchange, human resource exchange, physical resource exchange and events. The identification, organisation and representation of an interaction diagram are illustrated in Figure 6.8 where the basic interactions between the end user (e.g. Ford), the machine builders (e.g. Krause, Lamb Technicon, Cross Hüller) and component suppliers (e.g. Parker Hannifin, Rexroth) are described. The modelling notations and their meanings are described and explained in the diagram. The interaction diagram enables the basic interactions identified between the domain processes to be readily appreciated without presenting all of the details required for a complete analysis.

In Figure 6.8 for example, the end user raises a request for proposal and provide machine specifications to the machine builder. The machine builder, based upon this request and the specifications, presents the production design recommendation and their physical proposal (including machine control drawing resources) to the end user. The end user also has to provide the machine builder with parts drawings and parts resources for evaluation purposes. In terms of human resources, the end user sends its witness team and acceptance team to the machine builder to monitor the operation of the machine throughout the life-cycle, whilst the machine builders sends its installation team and training team to service the end user on site.

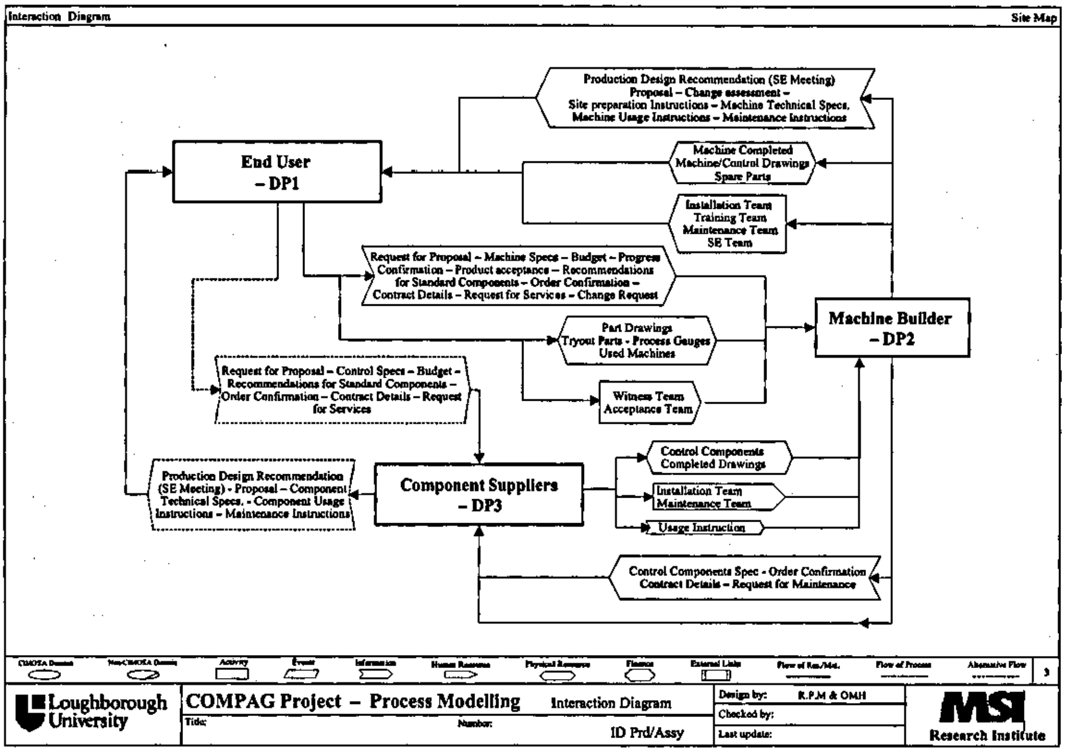


Figure 6.8 Interaction Diagram of domain processes (Source : Monfared 2002 [31])

Domain processes represent sequences of business processes and enterprise activities that are carried out to realise the objectives of the domain of interest. They have defined inputs and outputs. According to the CIMOSA modelling approach, a domain process could be decomposed to elementary units, business processes and enterprise activities. The decomposition of the processes is based on functional decomposition principle i.e. it is determined by i) the process of concern, ii) the focus of the modelling and iii) the modelling point of view, whether the process itself is a module which is reusable [134]. Figure 6.9 shows the individual business processes within a sub-domain process (e.g. Production Design). In Figure 6.9, the business processes of "Production Design" includes: "Administration", "Concept Design", "Design Review", "Detailed Design", "Documentation", "Assembly Shop Meeting" and "Commissioning".

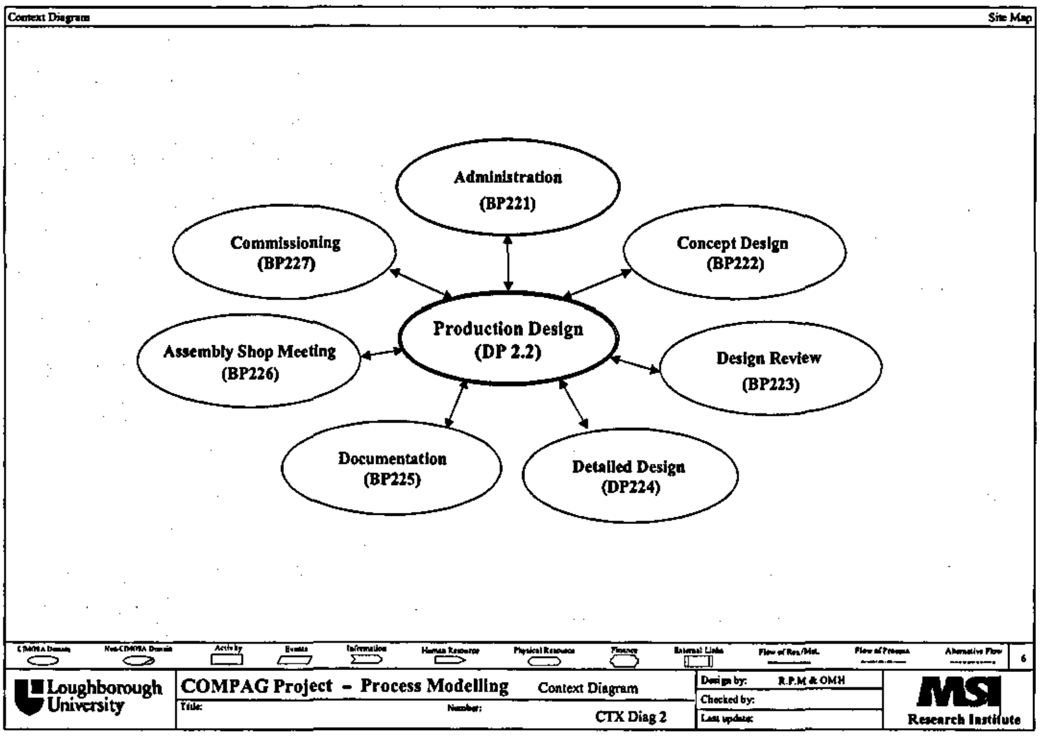


Figure 6.9 Business processes within a domain process

Each domain process, business process and enterprise process has an identifier, for example, the machine builder Cross Hüller is identified as DP2 (see Figure 6.7), sub-domain process "Production Design" as DP2.2 (see Figure 6.7), the "Concept Design" business process within "Production Design" is identified as BP2.2.2 (see Figure 6.9) and enterprise activity "Initial Layout Design" within the "Concept Design" business process e.g. as EA2.2.2.1 (see in Figure 6.10) and so on. The inputs and outputs of a domain process apply to enterprise activities; enterprise activities function to transform inputs to outputs, whereas business processes just concatenate the activities.

The decomposition of processes within the enterprise is represented in a structure diagram. A structure diagram organises domain processes, business processes and enterprise activities in a hierarchical manner. Figure 6.10 shows the structural decomposition of processes within the "Production Design" domain process of Cross Hüller. The business process "Concept Design" (BP2.2.2) comprises the "Initial Layout Design" (EA2.2.2.1), "Station Layout" (BP2.2.2.1) and "Advanced Planning" (BP2.2.2.2). Further decomposition illustrates that "Station Layout" consists of "Design Modules", "Design Tools/Holders", "Design Heads/Spindles", "Design Fixtures/Clamps", "Design Transfer/Nets", "Design Centre Base, Wing base" and

"Design Lubrication System". Consistency of the decomposition of each process is maintained by a unique identifier, placed in the top right hand corner of each rectangle.

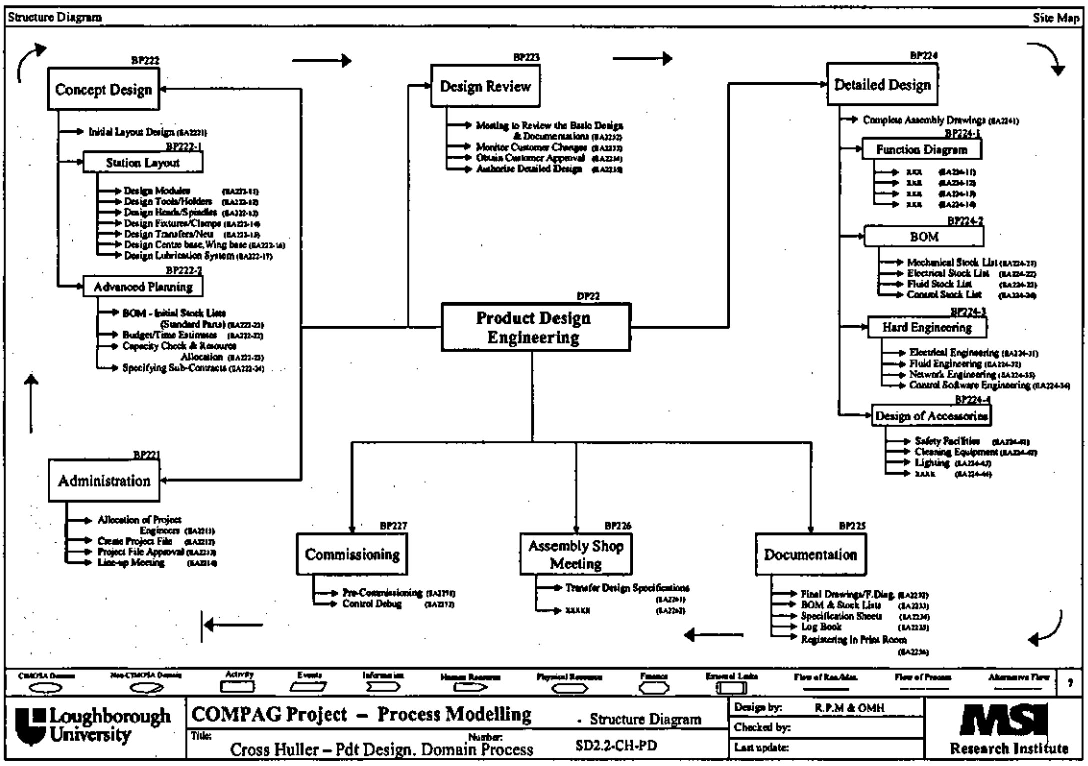


Figure 6.10 Structure Diagram

The fourth and last CIMOSA diagram in MMA is the activity diagram. An activity diagram shows the time sequence of business processes and enterprise activities (see Figure 6.11). Each business process and enterprise activity is represented by rectangular boxes. The flow of process from one business process / enterprise activity to another is represented by chained arrow-headed line. An alternate sequence of process flow is represented by a dashed arrow-headed line. The flow of information, physical resources, human resources and events is represented by continuous arrow-head lines. Unlike the interaction diagram, the information, resources and events shown in the activity diagram are specific to the business processes and the enterprise activities.

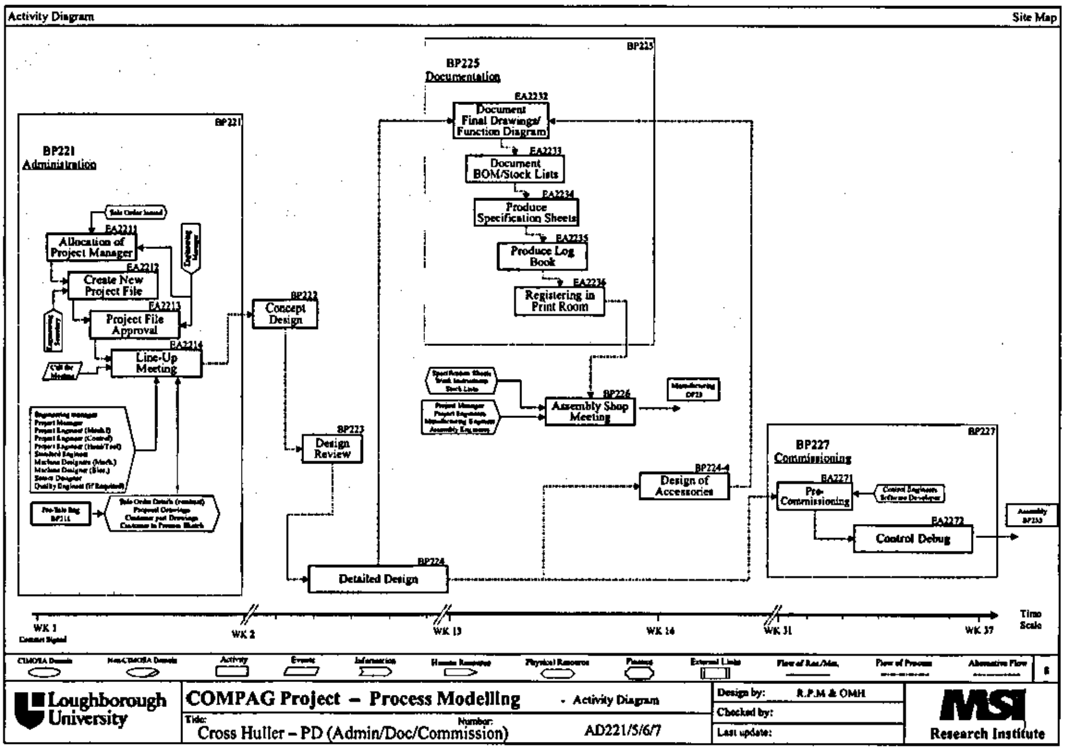


Figure 6.11 Activity Diagram

In Figure 6.11, the sequence of administration and documentation activities undertaken by the machine builder in the production design process are illustrated. The signing of the contract at week 1 marks the start of the project with the sales order being the physical resource that triggers the activities. The Administration process (BP2.2.1) entails the allocation of project manager (EA2.2.1.1) and with the help of an engineering secretary a new project file (EA2.2.1.2) is created. Once the approval for the project file (EA2.2.1.3) has been obtained from the engineering manager, a line-up meeting (EA2.2.1.4) is called to plan and discuss the project and flesh out the basic concept design (BP2.2.2). Personnel (i.e. human resources) involved in the line-up meeting (EA2.2.1.4) include, the engineering manager, the project manager, the project engineers for mechanical, control and head / tools, standards engineers, machine designers for mechanical and electrical systems, senior designers and quality engineers. During this meeting, details of the sales order (i.e. the contract) are released to those involved in the project. Proposal drawings, customer parts drawings and the in process sketches are disseminated. Following the project administration the concept design (BP2.2.2), design review (BP2.2.3) and detailed design (BP224) of the machine are carried out from week 2 to week 13. The detailed design of the machine is sent for documentation (BP2.2.5). Documentation covers the final approval of the mechanical layouts, fixtures and tooling and the function diagram which is a representation of the time sequence progression of the states of actuators and sensors within the machine. A bill of materials / stock list for the complete set of parts required for the machine is produced along with appropriate specification sheets. The project logbook is produced and all of the information registered in the print room where the information is stored for future reference. When the design has been registered in the print room (EA2.2.3.6), an assembly shop meeting (BP2.2.6) is being held. Project managers, project engineers, manufacturing engineers and assembly engineers will attend this meeting. The specification sheets, work instructions and stock lists are presented during the meeting and any issues addressed prior to passing the project on to Manufacturing (DP2.3).

Collectively, the four sets of diagrams can be followed from top-to-bottom, elaborating the picture and the complexity of the problem space. MMA is just a diagramming approach, implemented using graphical tool Microsoft Powerpoint [167] and Visio [168], producing "static models". However, there is a need to be able to simulate dynamically to evaluate in the process with respect to time, cost and the response to input scenarios.

6.2.5 Dynamic Modelling of Business Processes

The set of diagrams created in the MMA has been found to provide a good way of representing activities, resources and processes with a view to their organisational structure. These diagrams present a step-by-step understanding of how CIMOSA concepts can be depicted and implemented in a graphical form [105].

One of the ways to achieve business process analysis and optimisation is to use dynamic modelling (or simulation) centred on computer executable business process models [169]. Graphical simulation packages allow the modelling of business processes, enterprise activities and resources and capture their interrelationships so as to present a representation of processes that can be optimised or undergo what-if analysis. On developing such a representation, data can be input to the model and simulations can be run. The simulations can be used as part of the analysis process and the results used to indicate conditions under which a process can be operated effectively.

The modelling tool used in this work is iThink™ [170]. It has a set of basic constructs that can be used in a compatible way to those of CIMOSA [31]. This facilitates the translation of the CIMOSA diagrams into the required format in iThink™. The simulation results from iThink™ can be exported to external statistical applications (such as Microsoft Excel) which provides an opportunity to analyse and classify the simulation data so that it could be used for the assessment.

iThink™

iThink™, developed by High Performance Systems Inc, allows the developing of "mental models" of dynamic systems. The software was developed based on *Systems Thinking*, which seeks to identify how things interact with other constituents of the system (a set of elements that interact to produce behaviour) [171]. In system thinking, an attempt is made to look for events and patterns of behaviour, seeking the underlying systemic inter-relationships responsible for the patterns of behaviour and events [172]. The approach allows the modeller to take into account the large number of interactions present in complex systems. Mental models are representation in the mind of real and imaginary situations. Sometimes such a vision is clear if the interrelationships between entities are clear, however, as the relationships become complex and more factors are added to the picture, it becomes impossible to grasp the whole picture. This is when mental models become useful. They help the users to "plot" their thinking onto the paper and enable the whole picture to be appreciated.

Based on the concept of system thinking and mental models, iThink™ is a software tool that helps to map mental models onto formal models and helps check their exactness by numerating, simulating and improving them by analysis and communicating [173].

6.2.6 Process Modelling with iThink™

Although iThink™ is not a CIMOSA compliant tool it offers a set of general purpose and simple model building blocks that allow the modelling of resources and activities, and the resource and control flows associated with activities [170].

Three levels of modelling abstraction are provided: the interface level, the map/model level and the equation level. The interface level is the highest level, providing an overview of the model structure created. The map/model level is where the actual model is being created, i.e. a detailed representation of the activities. The equation level is the lowest level, where the mathematical equations associated with the models are located. These three different levels facilitate top-down approach to model development.

At the interface level, the main building blocks in iThink™ are the process frame, bundled flow and the bundled connector. The process frame allows the user to represent high-level processes and provides the navigation to the building blocks at the map/model level. An example of the process frame is shown in Figure 6.12. Four building blocks "Administration–Process", "Administration-Duration", "Administration-Information and Resources" and "Administration-Conditions" have been created, which allow users to determine at a glance that four sectors have been created in the model. Bundled connectors link the sectors and are represented by red tipped arrows in Figure 6.12. The bundled flow, the blue solid arrow in Figure 6.12, allows the representation of material flows between the sectors / processes in the model.

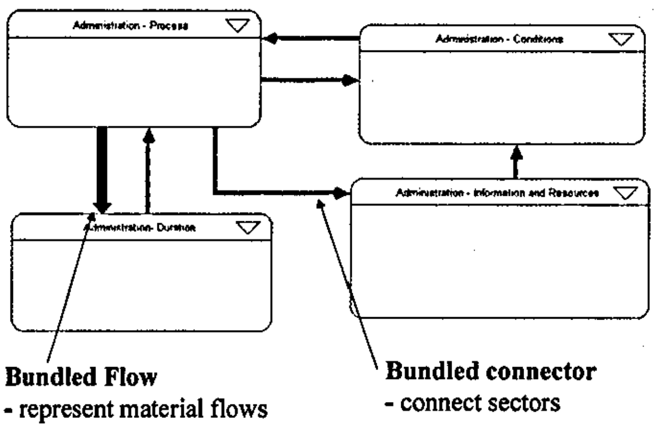


Figure 6.12 The bundled connector, represented as the red arrows

The map/model level is the level for constructing the detail within the models. At this level, the four main building blocks are: stock, flow, converter and connector, as shown in Figure 6.13. *Stocks* are accumulations which collect whatever flows into and out of them. There are different types of stocks but the default type is 'Reservoir', which is like a collection of undifferentiated piles of things [173]. Stocks accumulate the difference between inflows and outflows. A *flow* works like a tap, allowing inflow to and outflow from a stock. Each flow has an arrowhead to indicate the direction of the flow. The *converter* converts inputs into outputs; it could be used for depicting anything, it holds values for constants, defines external inputs to model, calculates algebraic relationships and serves as the repository for graphical functions [173]. The *connector*, which is a line with a node at one end and an arrow at the other end, serves to connect the model elements.

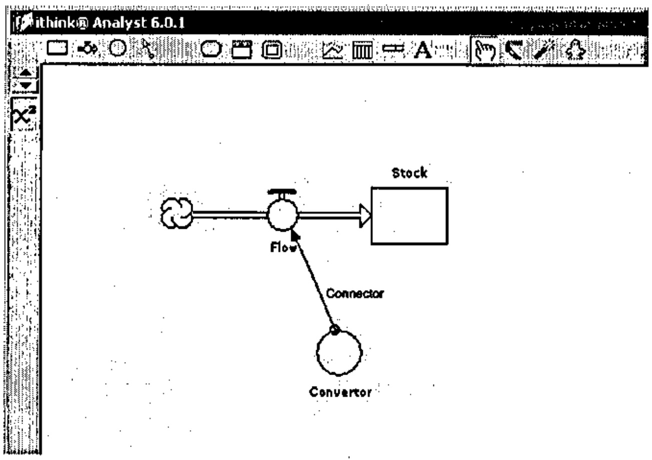


Figure 6.13 The basic building blocks in iThink™

One other useful construct is the sector frame, as shown in Figure 6.14. As the name implies, sector frames enable the creation of sectors for the grouping of functionally related elements in a model. In Figure 6.14, two sectors have been created for the "Administration" (BP2.2.1) (see Figure 6.9 and Figure 6.10, section 6.2.4 above) and the enterprise activities associated with the business process are grouped together using the sector frame. The creation of a sector frame at the model construction level will create a process frame at the interface level.

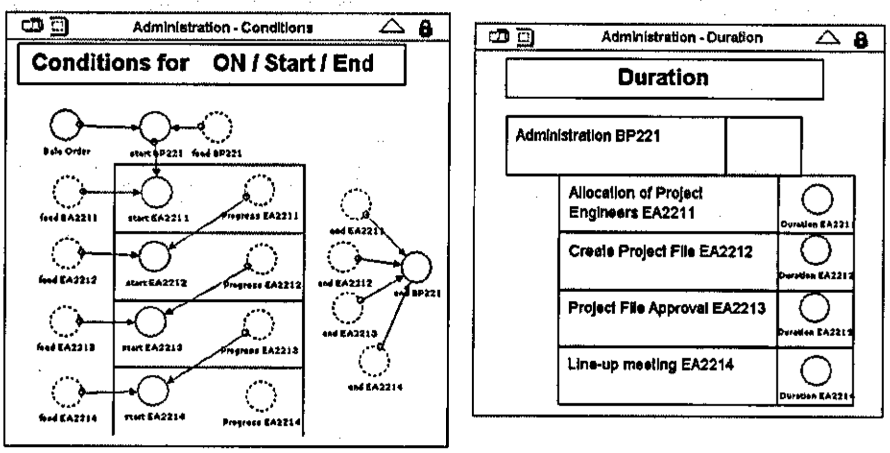


Figure 6.14 Sector frame in iThink

By using these building blocks, the static models created using the CIMOSA modelling approach are mapped onto iThink™ structures and basic constructs. To facilitate this process and ensure a high level of consistency and correctness in the translation from CIMOSA, four views have been created within iThink™: process, information and resources, conditions and duration. *Process* is used to describe the business processes and the enterprise activities. *Information and resources* show the inputs to the processes; these could be information, physical resources and human resources required for the processes. *Conditions* state the causal relation between and among the processes, stipulating when a process could start. For example, the ending status of one process is the execution status of another process or a process could start when the preceding process has progressed for a certain period of time. The *Duration* is to show the duration of each process, how long each process last. In this work, only the business processes and enterprise activities have been modelled. For example, in Figure 6.9, the DP "Production Design" is not created in the model; rather, its lower level activities are modelled. This is because the lower level processes and activities define the enterprise functionality (i.e. the things to be done) and have the largest amount of detail, crucial for model simulation.

The business processes and enterprise activities are categorised into two levels of processes. A process that can be further decomposed to more elementary processes is categorised as Level 1. A process that is already at its most elementary level is categorised as Level 2 processes. Figure 6.15 and Figure 6.16 show the translation of the business processes "Administration BP2.2.1" and one of its enterprise activities "Allocation of Project Engineers EA2.2.1.1" into iThink™ representations respectively. The basic constructs used in the models (e.g. start, running, duration, progress, end) will be explained in the later part of this section.

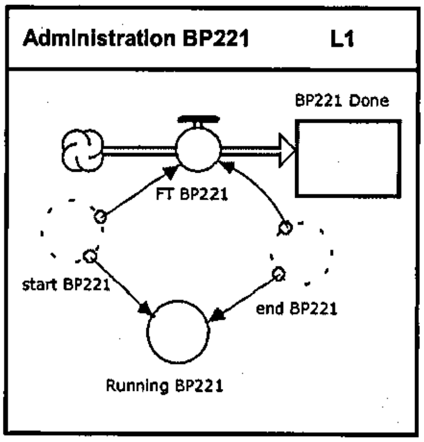


Figure 6.15 Business process 'Administration BP221' modelled in iThink™

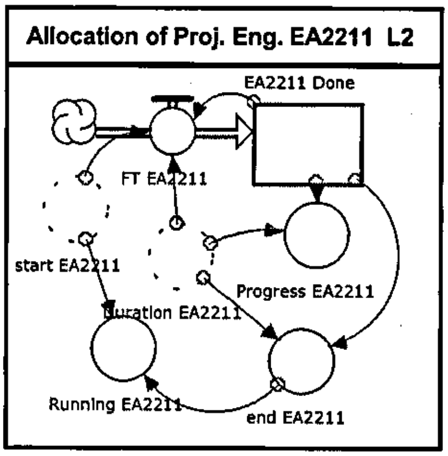


Figure 6.16 One of the enterprise activities under "Administration BP2.2.1", "Allocation of project Engineers EA2.2.1.1" modelled in iThink™.

The differences between the Level 1 process (L1-P) and Level 2 process (L2-P) are:

1. L1-Ps chain the execution of activities, they do not transform anything.
2. The input and output of L1-Ps are the summation of the inputs and outputs from L2-Ps
3. L2-Ps require resources and time for full execution
4. L2-Ps have function input and function output
5. L2-Ps have durations
6. L1-Ps end when their L2-Ps activities end.

The basic constructs adopted to represent a process are illustrated in Table 6.1. The "converter" has been found to be particularly useful since it can be used to represent resources and control the start and ending statuses of processes, as shown in Table 6.1 and later in Figure 6.17.

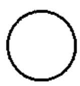
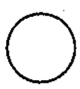
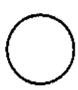
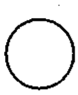
Constructs in iThink™	Activities modelled
BP221 Done 	The stock (i.e. reservoir) is used to represent an accumulation of time that is required to complete a task. When a task is completed, the reservoir will be filled.
 FT BP221	The flow controls the inflow into stock.
 start BP221	A converter is used to represent the start of a process. Resources that are required to start the process are connected to this converter.
 end BP221	A converter is used to represent the ending of a process. The conditions that are required to end the process can be connected to the converter.
 Duration EA2211	A converter is used to represent the duration of a process. This states the duration of a task, which is defined by the user.
 Progress EA2211	A converter is used to represent the progress of a process. It will show the percentage of work that is completed within the process.

Table 6.1 Constructs that are used to represent the activities within a process

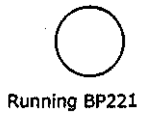
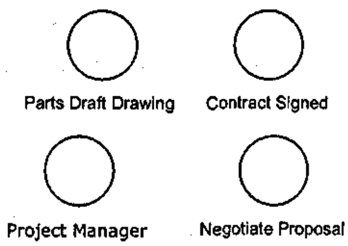
	This is an extra construct, using a converter to represent that a process is in progress, showing the starting and ending of a process. The difference between this and 'Progress' it just shows the start and end of a process, but not the amount of work that is completed.
	This represents a summation of the resources that is required for an activities or a process. The different resources are connected to this converter using connectors.
	The converters are used to represent the different types of resources that are required. These include physical resources such as "Parts Draft Drawing", events such as "Contract Signed", human resources such as "Project Manager" and information such as "Negotiate Proposal". These resources are connected to the "Feed" converter.

Table 6.1 (continue) Constructs that are used to represent the activities within a process

By using the constructs described in Table 6.1 to represent the activities and resources within a process, models can be developed to capture how and when an activity or process starts, the ending status, the duration and its progress. The main information that has been generated using these constructs under the process view is the time required to complete the task (i.e. collected in 'Stock') since this is the primary parameter required by the stakeholders when analysing the impact of any changes to system designs or business processes (see section 2.2). Each of the constructs in the models has an identifier associated with the activities directly mapped from the CIMOSA representations (see section 6.2.4).

The physical, human and information resources that are required by a BP or EA are defined in the "Information and Resources" section. This is illustrated in Figure 6.17. Converters are used to represent the individual resources that are required and also to represent a summation of the total resources that are required as input to a process, as illustrated by converters named as "Feed". For example, "Feed EA2.2.1.1" will "contain" all the resources requirement for BP2.2.1; the resources that are required for EA2.2.1.1 (i.e. Feed EA2.2.1.1) are a "Sale Order Issued" and the availability of an engineering 'Engineering Manager'.

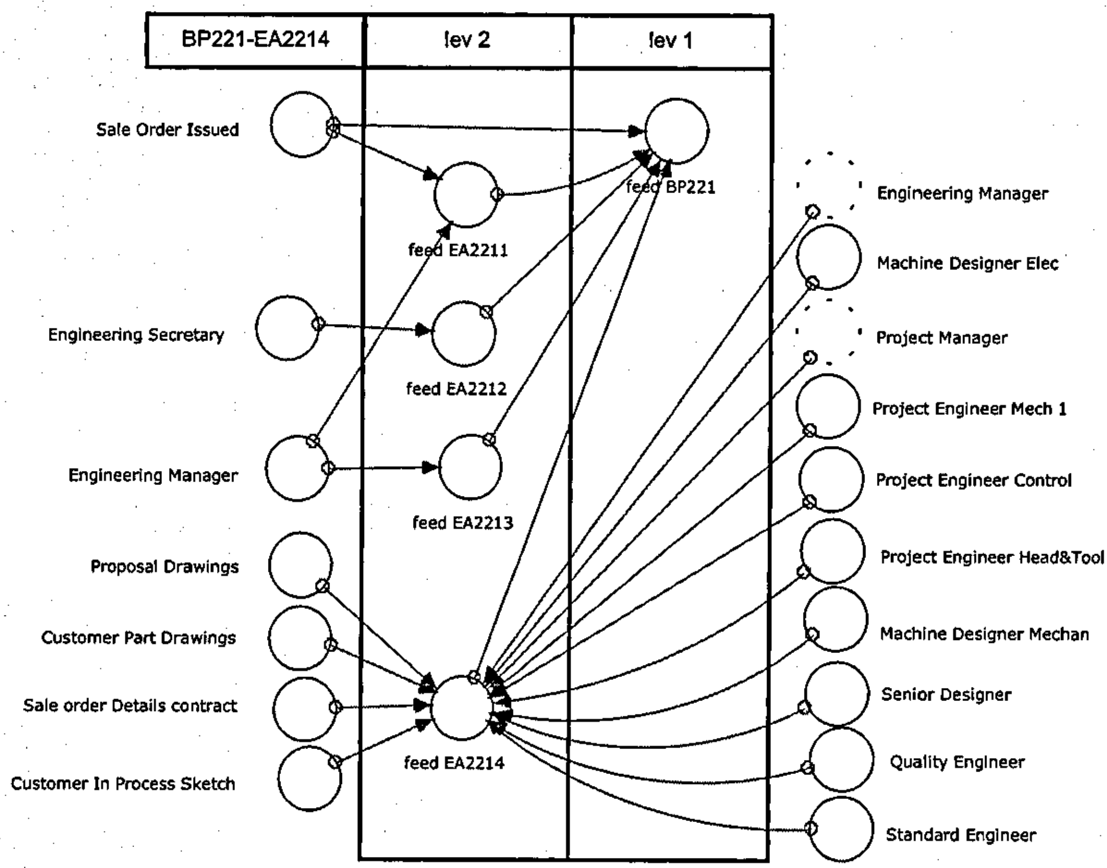


Figure 6.17 The Information and Resources section illustrating the inputs to EAs and BPs. The converter named as "Feed" is a summation of all the resources that is required as the input to an EA or a BP. Note that the information and resources required for a BP is a summation of that required by its EAs.

Figure 6.18 shows the conditions which specify the triggering and ending of BPs and EAs. The process "BP2.2.1" starts when it has the necessary resources (i.e. "feed BP2.2.1" and "Sale Order Issued" (see Figure 6.17)) and "EA2.2.1.2" starts when all the resources are available (i.e. feed EA2.2.1.2) and 'EA2211' is in progress (i.e. represented by "Progress EA2.2.1.1"). The use of the "Progress" construct allows the user to have the flexibility of specifying whether the process should start when the preceding process is totally completed (i.e. 100%) or when it is still in progress (i.e. starts when the preceding process is 50% completed). The ending status of an EA is determined by its duration, i.e. the time allocated for the activities. The BP, on the other hand, ends when all its EAs are completed. For example, in Figure 6.18, EA2.2.1.1, EA2.2.1.2, EA2.2.1.3 and EA2.2.1.4 have to be completed prior to the end of BP2.2.1. The triggering conditions of EAs require i) causal relations (i.e. starts when an activity has finished) and ii) resource inputs. The BPs are triggered by causal relations i.e. starts when another BP has ended. The beginning of a BP (i.e. L1-P) will start the associated EAs (i.e. L2-Ps). For example, the beginning of "Administration BP2.2.1" will trigger the EA "Allocation of Project Manager EA2.2.1.1" (see Figure 6.18).

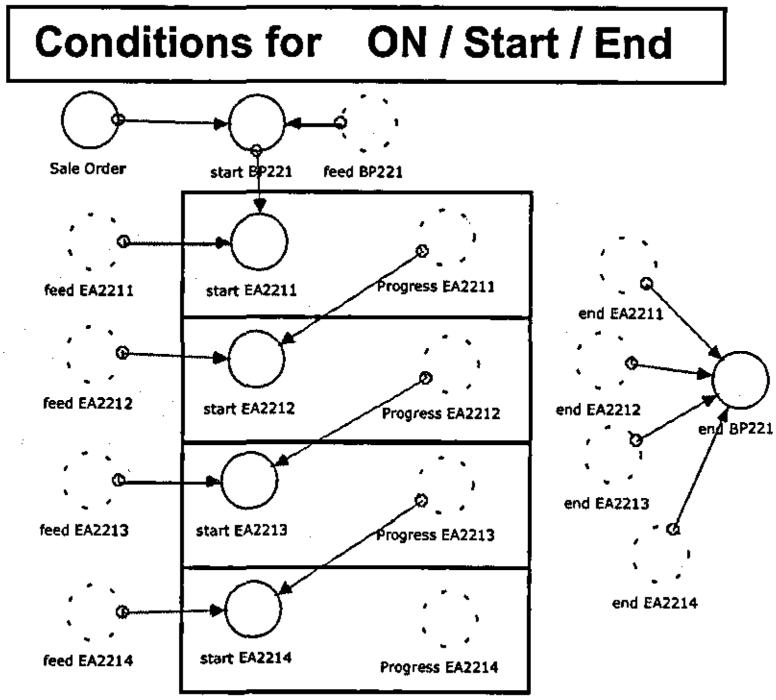


Figure 6.18 Conditions which trigger the starting and ending of the process.

The times taken to complete the BPs and EAs are represented using a "duration" converter, as illustrated in Figure 6.19. The duration converter contains a value that depicts the amount of time allocated for the activities. The duration of a BP is the total time required for completing the employed EAs.

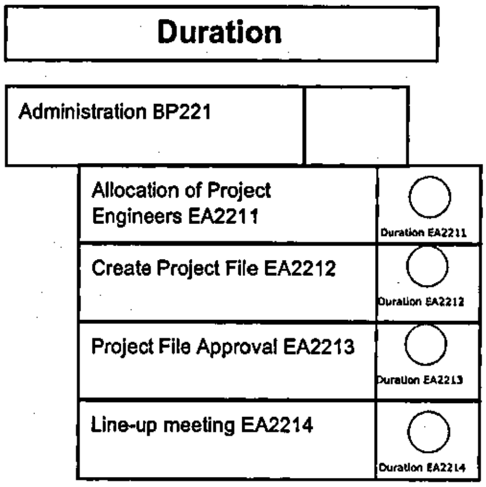


Figure 6.19 Duration of the activities is represented using converters

It can be seen from the above descriptions that the four views created in the dynamic modelling (i.e. process information and resources, conditions and duration) facilitated the translation of the static CIMOSA diagrams to iThink™ models. By having such a structure, data consistency during the translation can be maintained. In addition, a modular representation of processes in iThink has enabled the models to be reused easily. This helps to reduce the time taken and the effort required to construct an entire model for the enterprise.

The main purposes of the CIMOSA and corresponding iThink™ developments was to model the dynamic functionalities of the enterprise, i.e. develop a formal dynamic functional model of an enterprise. The model has not designed to illustrate how the information, human resources or physical resources are obtained, just how the resources support the activities and functions behind the design and implementation process of the automotive engine production machinery. If required in the future, information and resource models can be readily developed and included in the modelling activity by using the basic modelling constructs outlined above. However, since the focus is on functionality, the information, resource and organisation components of the model have not been developed in this work.

6.3 Modelling of the CCG Process

The MMA constructs outlined above have been used to define static models of the processes behind the implementation of the CBA systems in the CCG projects. As mentioned in Chapter 4, the CCG projects identified four different environments to support the control, engineering, 3D modelling and Human Machine Interface (HMI) system functions. The CBA process model has been used to identify the processes involved in using the CBA approach in the CCG projects, such as how a machine would be configured, how a component is created and how VRML 3D models are created. Arguably, these processes would form only a part of an enterprise's business processes or activities, but as standalone items, they can be used to indicate the main processes involved in the development of a CBA system. There is a need to focus on these processes in isolation so that the industrialists can appreciate the activities that are involved in producing CBA machines and assess the impacts on their current way of working in isolation.

The CCG project has been chosen as the domain of study i.e. the "CB system" (see Figure 6.20). The complexity of the project is broken down and organised into sub-domains, as shown in Figure 6.20. The domains which are not studied within this research are identified as the non-CIMOSA domains, i.e. "Common Database", "Control Environment" and "Engineering Tool". These domains have not been studied because they are mainly activities undertaken to generate the tools that are expected to be supplied to the stakeholders i.e. they would not have to be redeveloped. Domains that are within the scope of study in this work are the CIMOSA domains (e.g. "Create Machine", "VRML 3D model", "Create Components", "HMI", "Machine Simulation" and "Machine Monitoring") since these are the activities that any company adopting the approach would have to undertake and it is important that they are understood in detail. This context diagram outlined in Figure 6.20 defines the scope of any CB system and indicates the typical functionalities. The process involved in each of the domains has been further defined using the structure diagram to illustrate the decomposition of processes within each DP. A detailed description of each of the DPs is given in Appendix A. A brief summary is listed below.

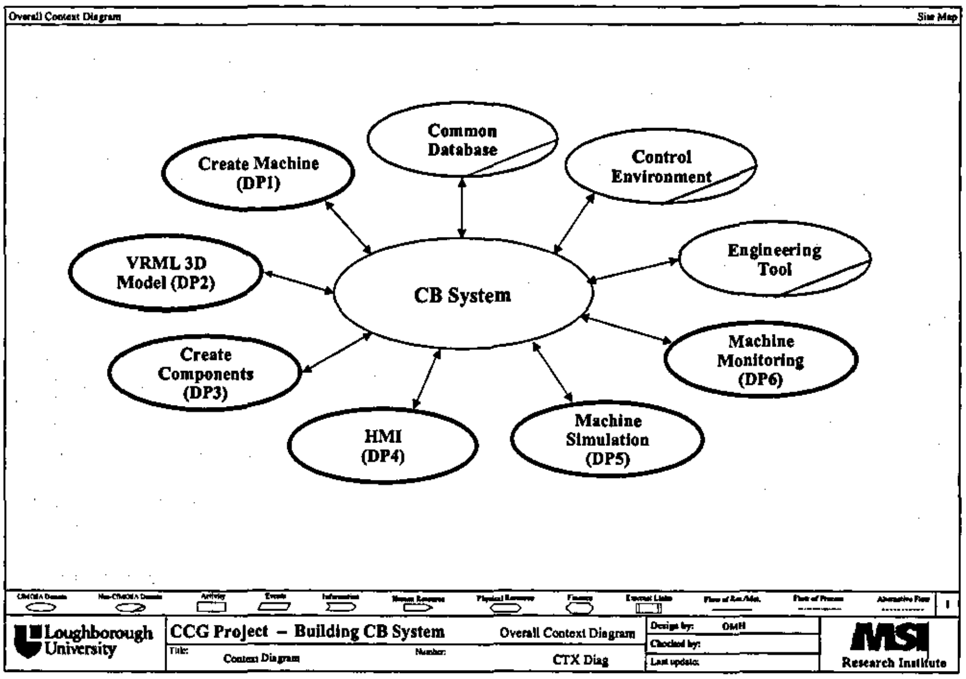


Figure 6.20 The overall context diagram for creating a system

Create Machine DP1

A structure diagram representing the activities that are involved in the "Create machine" DP1 (using the Process Definition Environment (PDE) software tool created by the CCG projects) is shown in Figure 6.22. Within this domain process, two business processes have been identified, namely "Machine Configuration" and "Test and Debug". Machine Configuration describes the processes involved in configuring a machine. A machine is created by first defining a system (i.e. Create System BP1.1.1) in the PDE, in a Machine Hierarchy window (i.e. the Demo Machine system), as shown in Figure 6.22. Each machine system is subsequently decomposed into sub-systems (i.e. Create Subsystem BP1.1.2), which is followed by defining the sub-systems either based on the components (i.e. Components View BP1.1.3) or based on the modules (i.e. Modules View BP1.1.4), as shown in Figure 6.23. The component hierarchy indicates that the "Demo Machine", "Sub System A" is made up of a "Lamb Clamp" and a "Lamb Parts Seated Unit", each of which is composed from the list of components in the Component Catalogue (Figure 6.22), i.e. represented by EA "Add components EA1.1.3.1". The "Lamb Clamp" is composed of a "Lamb Clamp Actuator" element whereas the "Lamb parts Seated Unit" is composed of four basic two state sensors and a "Part Seated sensor". The interlocks are added (EA1.1.3.2) to the states of the components. In Figure 6.22, the interlock "Sub System A\Lamb Parts Seated Unit\Sensor 1\State 2" is added to the "Home" state of the Lamb Clamp Actuator. The Module Catalogue View (see Figure 6.23) illustrates the catalogue of components that have a VRML representation associated with them, i.e. Add Geometrical modules EA1.1.4.1 (see Chapter 4 for details of the component based system architecture). The Machine Hierarchy module view (see Figure 6.23) indicates that the "Demo Machine", "Sub-System A" is composed of a "Lamb Clamp Module" and a "Lamb Fixture". The link point of each of the modules are selected (EA1.1.4.2) and the interlocks are added (EA1.1.4.3) as in the case of using "Component views".

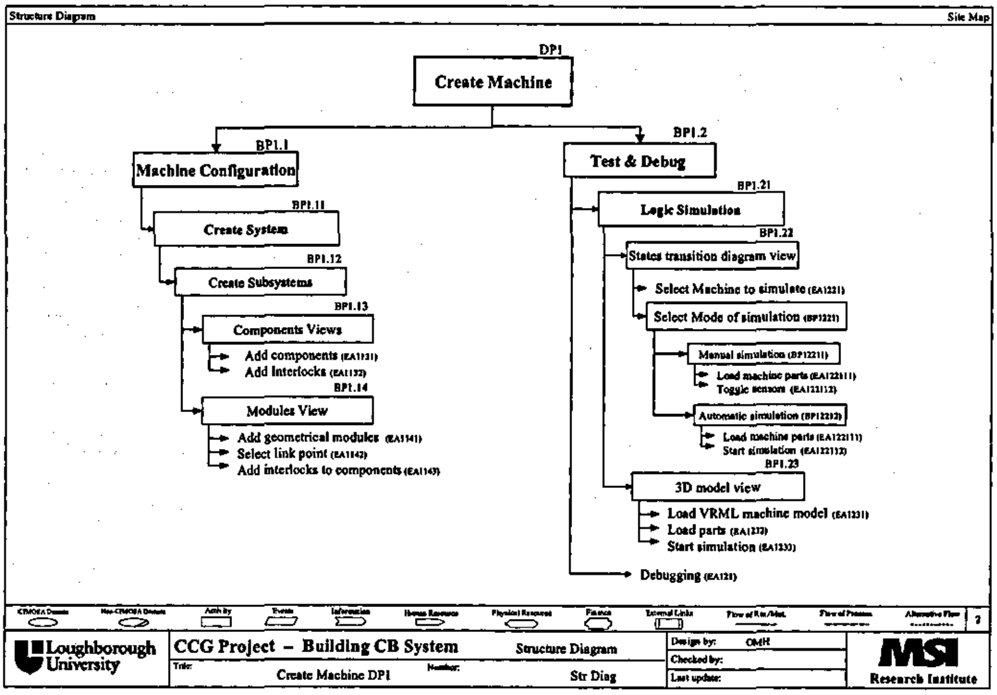


Figure 6.21 Structure diagram for 'Create Machine '

The activity diagram can be used to illustrate the basic activities and the resources that are required for "Create Machine" process (Figure 6.24). For example, for the BP "Machine Configuration BP1.1", the human resources required are the Engineers; the information required for this process is machine structure, the components list and the machine interlocks; the physical resources input are the timing diagrams (see Chapter 4), the PDE engineering tools and the common database. The information that is produced from this process would be machine, machine subsystem and machine modules. These form the input resources to the BP "Testing and Debug BP1.2". If the components and / or modules are not already available in the respective catalogues then the "Create Component" (DP3) and "VRML 3D Model" (DP2) processes (see below for details) have to be undertaken.

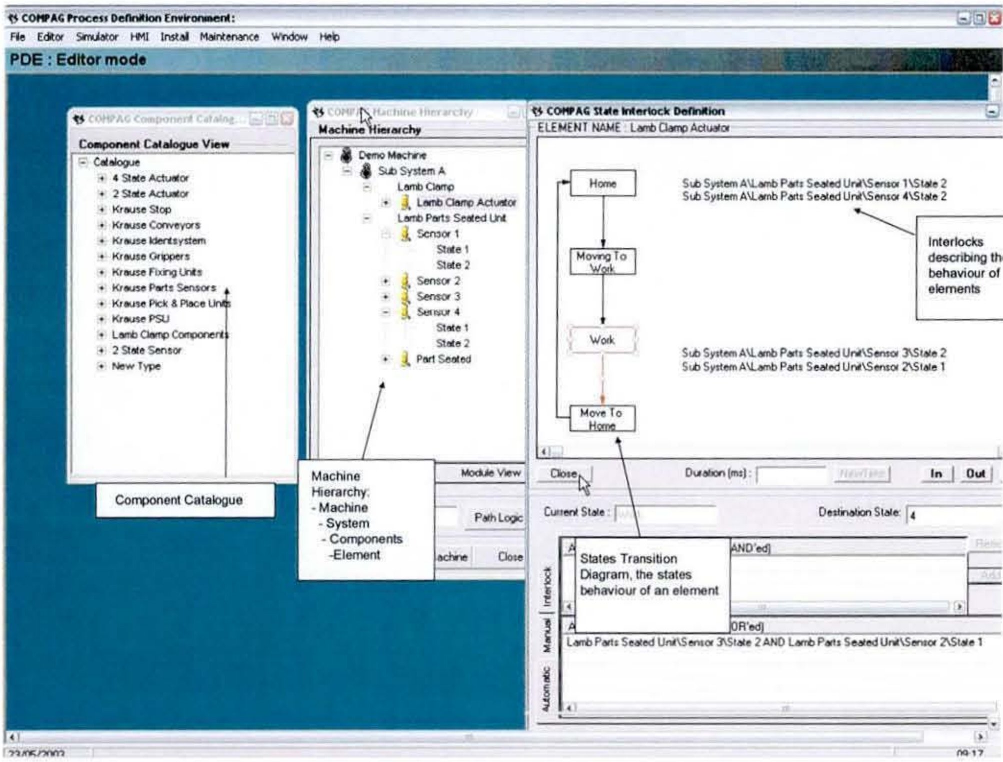


Figure 6.22 Machine creation using the PDE

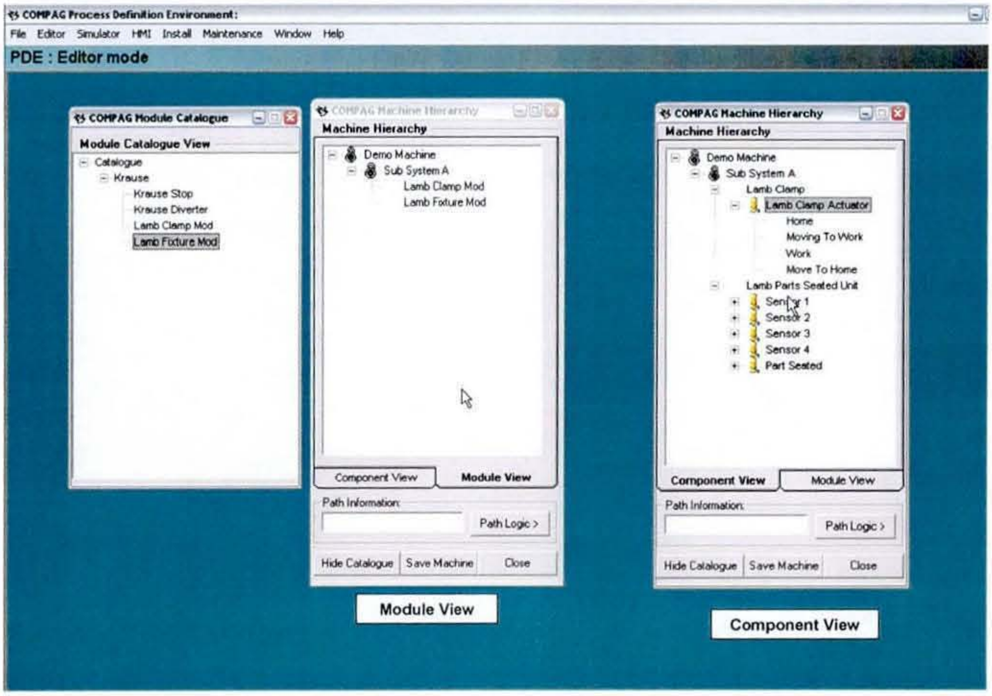


Figure 6.23 The creation of subsystems from the module view or the component view in the PDE

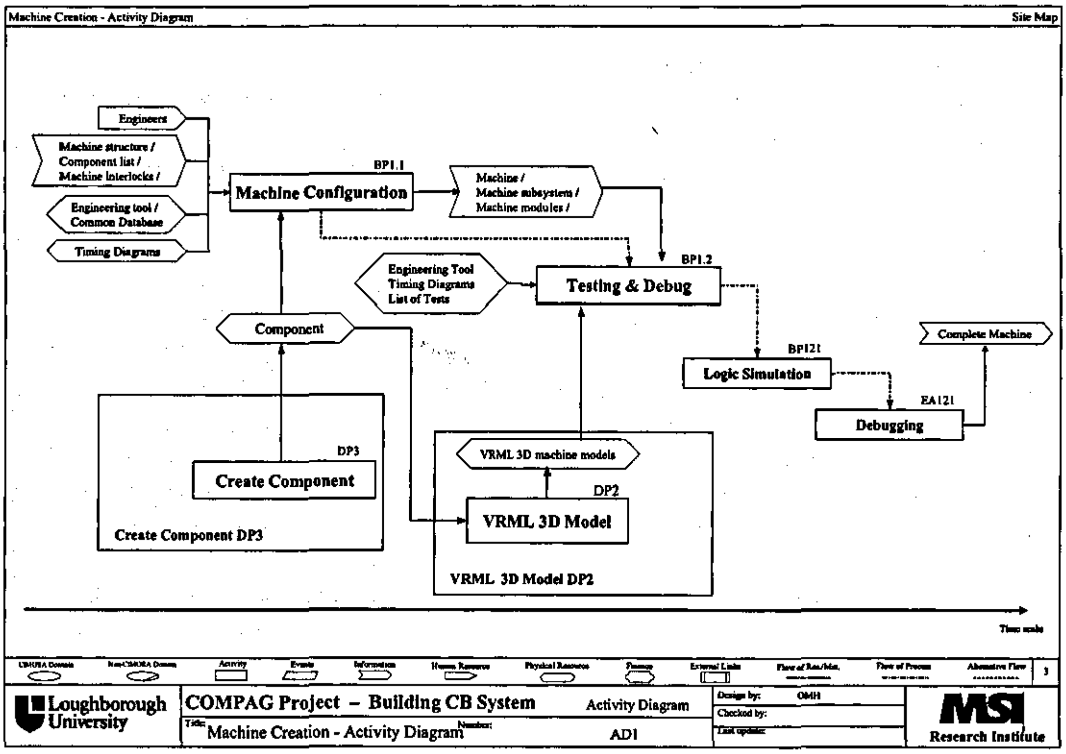


Figure 6.24 Activity Diagram showing activities in creating machine

Create VRML 3D Model DP2

A structure diagram to illustrate the basic business processes and enterprise activities that are undertaken in creating a VRML 3D model is illustrated in Figure 6.25. There are six main BPs involved. The first process is to "Analyse and understand the object to model" (BP2.1) "in terms of its geometry, behaviour and dynamics" and secondly to "Define the architecture elements" (BP2.2). The architecture elements are elements which form the basic structure (i.e. motion dynamics, state progression, conditions when states change, link points to indicate the physical locations where elements can be joined to form modules, view points and user interface buttons to allow the VRML model to be animated in response to state changes prior to configuration within the machine structure) for creating a VRML model. These two processes, (i.e. BP2.1 and BP2.2) have been identified as alternative flows (i.e. dashed arrows in Figure 6.25) because these processes only take place if a library of VRML modules does not exist. If such a library is already available, the "Create VRML 3D model" process will begin with "Specific Case Description" (BP2.3). This is a process that requires the engineer to know the specific requirements of the object to model, such as understanding the mechanical parts, the moving parts and the modelling requirements in terms of the level of detail require, the information or data required and the user expectations of the above. Following this process the basic shapes are created (BP2.4). If at this stage, CAD (Computer Aided Design) files are already available, the shapes can be directly exported in VRML format, if not, the basic shapes have to be defined in CAD and exported to VRML [13]. "Component Configuration" (BP2.5) is the process for populating the various elements with the capability required for the CBA. This process includes populating the dynamic/static elements in term of their positions / orientations and kinematics (i.e. translations, rotations and movement routes), states (i.e. names, positions, colours (if sensors)), conditions (i.e. names, routes and default time taken), link points and viewpoints. When the components had been configured, they can be assembled (i.e. Component Assembly BP2.6) in terms of their geometrical appearance (i.e. floor space, selected components to be linked and link points on each component) and logical behaviour (i.e. behaviour when conditions and interlocks are set).

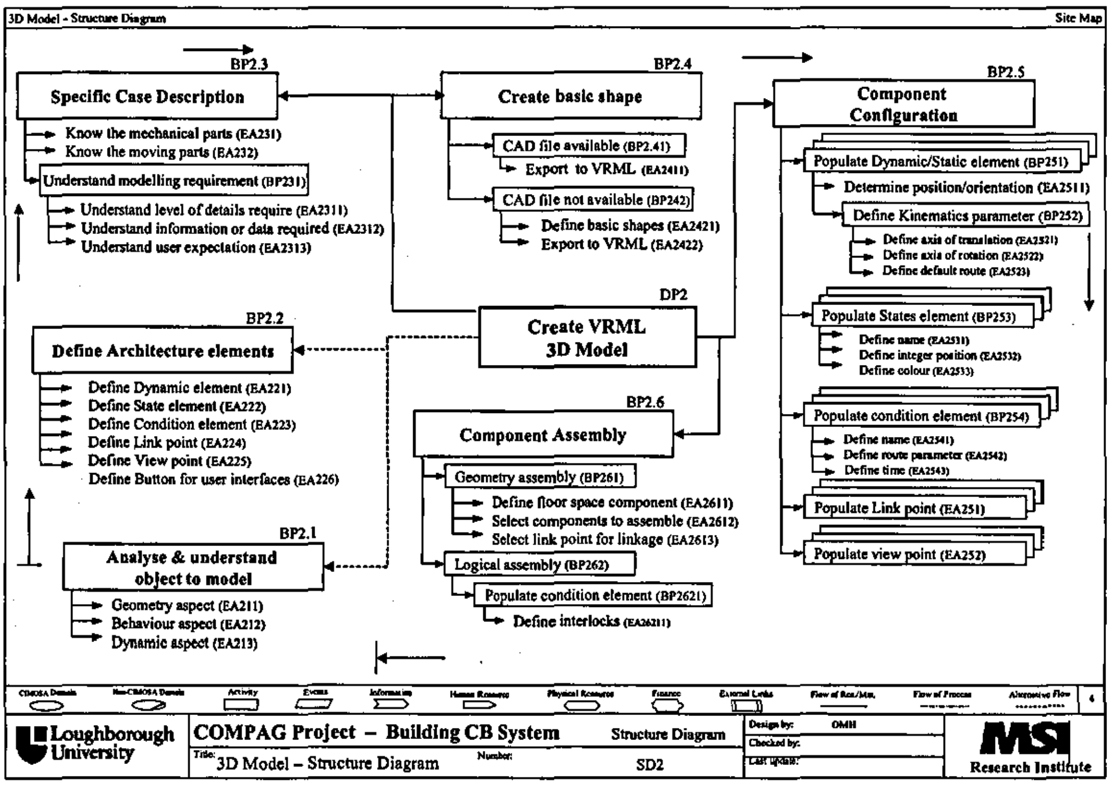


Figure 6.25 Structure diagram for creating VRML 3D model.

[13] Note: the VRML models used in the CCG process have been structured within an object oriented architecture to ensure that the functionality and composability required within the CBA can be achieved (Vera 2004 [135]).

The activity diagram in Figure 6.26 shows the sequential processes involved in creating VRML models. In addition, the diagram also illustrates the specific information and resources required during the processes. For example, the customers need to provide information on the type of machine logic, basic geometry, orientation, kinematics motion, parent/child relationship and logical description of states for the process "Analysis and understand object to model" to proceed. The customers will also provide their specifications, information on the purpose of modelling (i.e. visualisation or mechanical analysis) and the required level of detail during the "Specific Case Description" BP. Physical resources such as pictures or drawings of items to be modelled, machine layouts, state description of machine behaviour, mechanical movement, moving parts behaviour (i.e. range of movement, accelerations, velocities) and the basic shapes of the actuators will also be provided at this stage (BP2.3). The duration required for the activities has not been shown on the activity diagram because the duration of the processes depend on i) the size of the system and ii) the level of details required from the model.

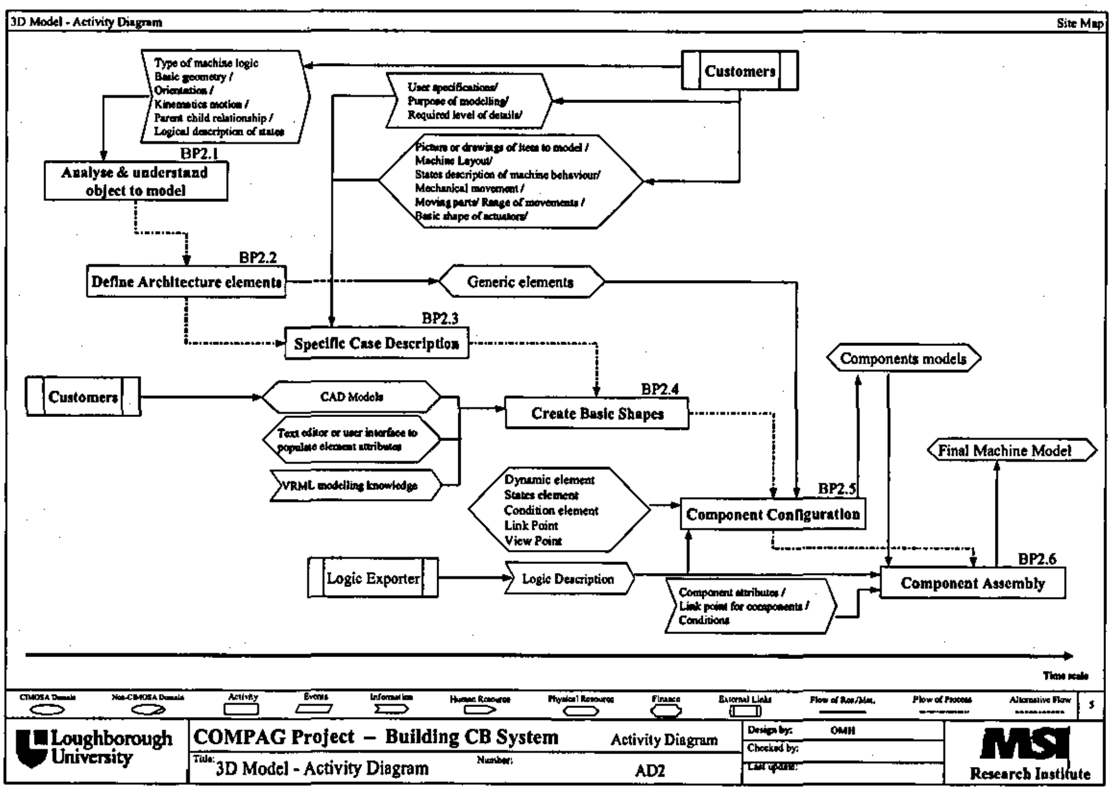


Figure 6.26 Activity diagram for VRML 3D model

Create Component DP3

The structure diagram in Figure 6.27 provides a structural decomposition of the processes involved in creating a component (DP3). In addition to the VRML 3D model DP, create component is a domain process that is of concern to the component suppliers when implementing CBA since end users and machine builders will expect commercial off the shelf components to be provided by these enterprises with the required functionality and visualisation capabilities. Without this knowledge it would be difficult for component suppliers to appreciate the effort required to generate the components and hence determine appropriate pricing structures. It is also important for the end users and machine builders to appreciate this effort in order for them to appreciate the increased cost for the increased capability that their systems will appreciate.

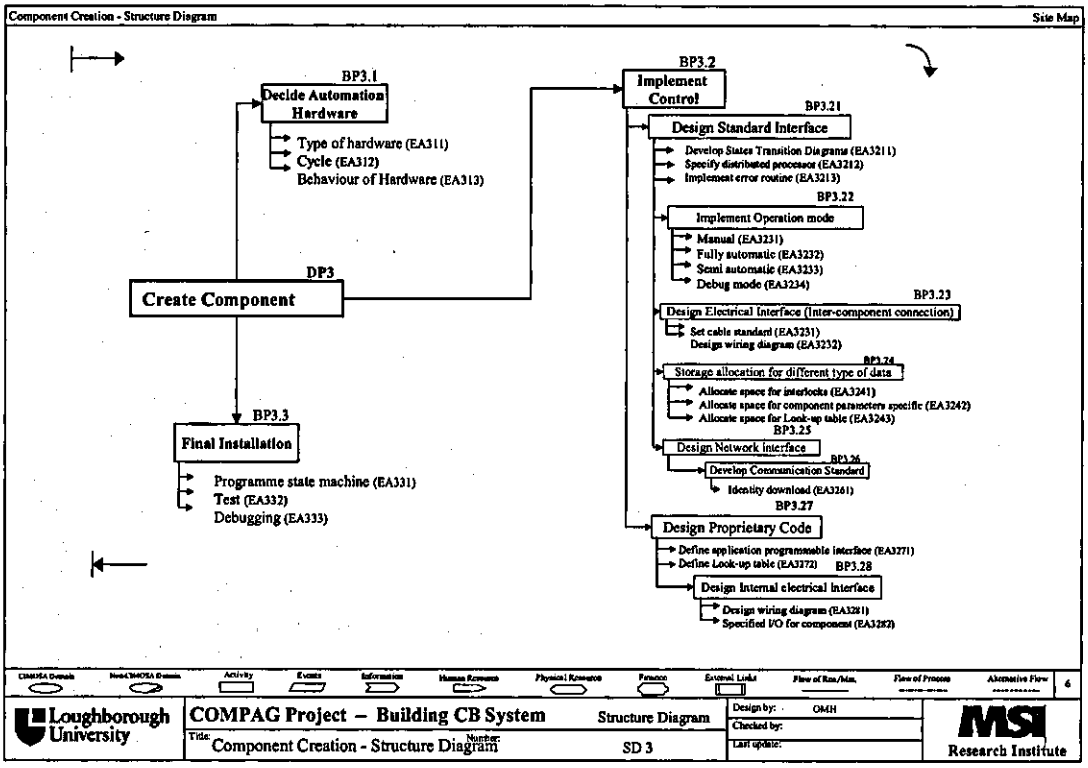


Figure 6.27 Structure diagram for creating a component

In creating a component, the automation hardware has to be determined (BP3.1), which will include activities such as determining the type of hardware (EA3.1.1), the cycle (EA3.1.2) and the behaviour of the hardware (EA3.1.3). The control implementation process (i.e. Implement Control BP3.2) comprises designing (i) a standard interface (BP3.2.1) and (ii) proprietary code (BP3.2.7). The activities involved in designing a standard interface include: i) developing state transition diagrams (EA3.2.1.1), ii) distribution services (EA3.2.1.2), iii) error routines (EA3.2.1.3), iv) Implement Operation Modes (BP3.2.2) (i.e. manual, fully automatic, semi automatic and debug), v) Design Electrical Interface (Inter-component connection, BP3.2.3) (i.e. cabling standards and design wiring diagrams), vi) Storage allocation for different type of data (BP3.2.4) (i.e. allocate space on processor system for interlocks, specific component parameters and state transition look up tables, and vii) Design Network interface (BP3.2.5) (i.e. identify an appropriate communication standard and identify the download capabilities). In designing the proprietary code, the application programmable interface (EA3.2.7.1) and a "look-up table" for the control interlocks (EA3.2.7.2) have to be defined. The internal electrical interface (BP3.2.8) has to be designed in terms of the wiring diagrams and specified physical inputs and outputs (either digital and / or analogue) for the component. Finally, the component is ready for final installation (BP3.3). During this process, the state machine is programmed (EA3.3.1) and testing (EA3.3.2) and debugging (EA3.3.3) are to be carried out.

An activity diagram has been constructed to illustrate the sequential order of the processes involved in creating a component (see Figure 6.28). Information and resources specific to each process have been highlighted in the diagram. From this diagram, it can be seen that the human resources involved includes mechanical engineers (to decide upon the automation hardware, the movement profiles and the I/O parameters and error recovery requirements), electrical and control engineers (both involved throughout the remaining processes and activities). Perhaps the most time consuming activities are during the process of implementing operational modes (BP3.2.2) where all possible combinations of manual movement and the types of errors have to be considered.

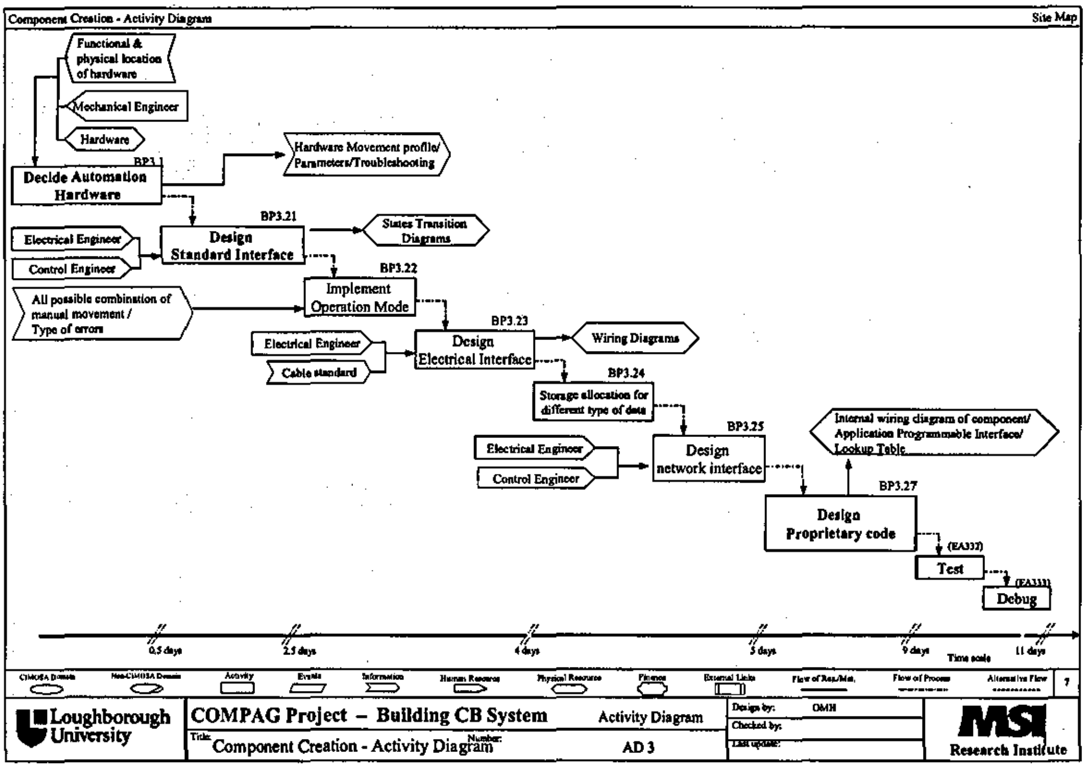


Figure 6.28 Activity diagram on creating a component

Based on the structural decomposition of the processes involved in creating a component, an estimation of the duration of each process is tabulated as followed (assuming that each working day has eight hours):

Process	Duration (Days)
Decide Automation Hardware BP3.1	0.5
Design Standard Interface BP3.2.1	2
Implement Operation mode BP3.2.2	0.5
Design Electrical Interface (inter-component connection) BP3.2.3	1
Storage allocation for different type of data BP3.2.4	0.5
Design Network Interface BP3.2.5	0.5
Design proprietary code BP3.2.7	4
Final Installation BP3.3	2
Total	**11**

Table 6.2 Duration of the processes involved in creating a component under the CBA

As shown in Table 6.2, it is estimated that it would take 11 days to implement a CBA component. It should be noted that this is the duration to implement the *first* component, from the process "Decide Automation Hardware BP3.1" to "Final Installation BP3.3". For subsequent components, the developer would not have to go through the complete process; the processes involved will be: Decide automation hardware BP3.1, Design Standard Interface BP3.2.1 (i.e. develop states transition diagram EA3.2.1.1), design electrical interface (inter-component connection) BP3.2.3 and design proprietary code BP3.2.7. As such, the duration for developing subsequent components will be 6.5 days as shown in the following:

Number of days required = (BP3.1)+(EA3.2.1.1)+(BP3.2.3)+(BP3.2.7)
= 0.5 days + 1 day + 1 day + 4 days
= 6.5 days

Therefore, the duration required to develop *n* number of components will be:

Number of days required = Duration for developing 1st component + (*n*-1)(Duration for developing subsequent components)
= 11 days + (*n*-1)(6.5 days)

For example, if there are 10 components to be developed from scratch, the duration will be as followed:

Number of days required = Duration for developing 1st component + (10 -1)(Duration for developing subsequent components)
= 11 days + 9 (6.5 days)
= 11 days + 58.5 days
= 69.5 days

Although the development effort seems high, however, research on CBA in the software industry has found that developing a reusable component requires three to four times more resources than developing a component for particular use [174]. This is mainly because for components which are designed to be reused, they have to be sufficiently general to cover the different aspects of their usages [175]. Therefore, the more the components are being reused, the higher the return of investment. In the automotive sector, the CBA components have to be designed such that they are generic to be used in a wide range of applications, for example, a component can be used in different types of machine, assembly or transfer line machine. To the component suppliers whose main business is to develop components, the development effort will be justified by the sales of the components. To the machine builders who have already been reusing most of their machine designs, the initial investment in developing a reusable component will definitely be justified by the high reuse of previous design (i.e. 80% of machine designs are reused) and thus the savings in the resources required.

Create HMI DP4

Figure 6.29 shows a structure diagram for creating a HMI. The processes involved are: i) understand user requirements (BP4.1) (i.e. identify the user, identify the usage, determine the information to display and determine the format of the visualisation), ii) understand operation procedure (BP4.2) (i.e. understand the architecture of the system, the prioritisation of information, manipulation requirements and level of manipulation required by each user), iii) define HMI pattern (BP4.3) (i.e. determine HMI functions, structuring of information, strategy for manipulation and action, diagnostic information display strategy and navigation mechanisms), iv) designing HMI (BP4.4) (see below), v) HMI configuration (BP4.5) (see below) and vi) Test and Debug (BP4.6) (i.e. test for automatic operation, manual operation, diagnostic messages). The major characteristics about the HMI creation under the CCG project is that a template is created which enables the generated HMI to be consistent, having a common look and feel (see section 4.2.3). The creation of the template (BP4.4.1) includes i) determining the fixed aspects (BP4.4.2) (i.e. the layout of the screens and the core view / commonality between screens), ii) determining the variant aspects (BP4.4.3) (i.e. determining the user identification, screen colours and font types) and iii) creating the graphics (EA4.4.1) that are to appear on the HMI. The template is used for HMI configuration (BP4.5), which is the process undertaken to populate the HMI with components that make up the machine. This is done by using the configuration functions in the PDE (section 4.2.3) and involves configuration of the HMI server machine, specifying the display preferences for each screen (e.g. images, colour, fonts) and specifying the layout of the pages. The completed HMI is then subjected to testing and debugging (BP4.6).

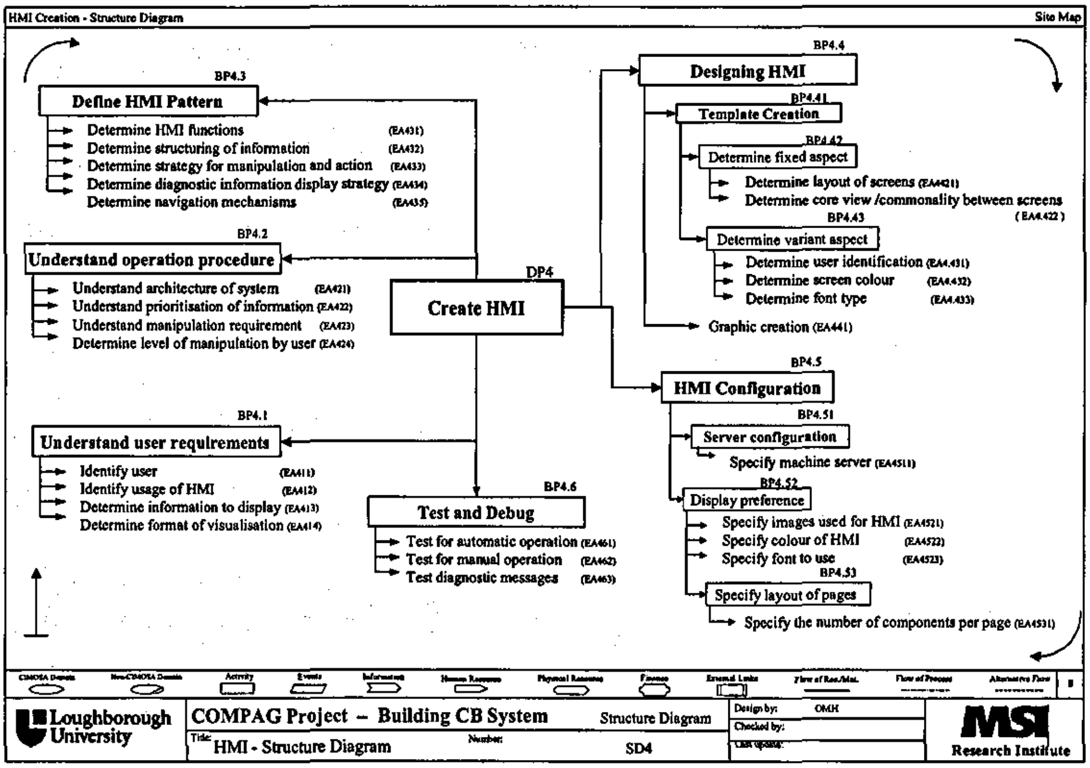


Figure 6.29 Structure diagram for creating HMI

The sequence of activities involved in creating a HMI and the required resources are as shown in the activity diagram in Figure 6.30. Information from the customers is important at the early processes so as to understand their requirements in terms of users and the operation procedures of the machine. The customers provide information such as typical usage scenarios of the HMI, what information to display and how to display it and the range of users of the HMI to enable an understanding of their requirements (BP4.1). In addition, they have to provide information on the machine operation sequence, logic associated with the control buttons, the sequencing of screens, the interaction between screens, the different information displays required for different users, the machine type and the machine architecture so that the HMI pattern can be defined. During the configuration of the HMI, the layout of the pages is specified and a configuration tool is required to configure the HMI in terms of i) the layout, ii) the HTML files that comprise the different HMI screens, iii) associated style sheets, and iv) included VRML 3D models. As in the case of the CBA processes that create VRML 3D models and components, testing and debugging of the HMI is the final business process to be carried out. As in the case of creating the VRML 3D model, the duration for the HMI creation activities has not been shown in the activity diagram because it depends on i) the information to display, ii) the graphics, iii) the interactive element on the screen and iv) navigation requirement.

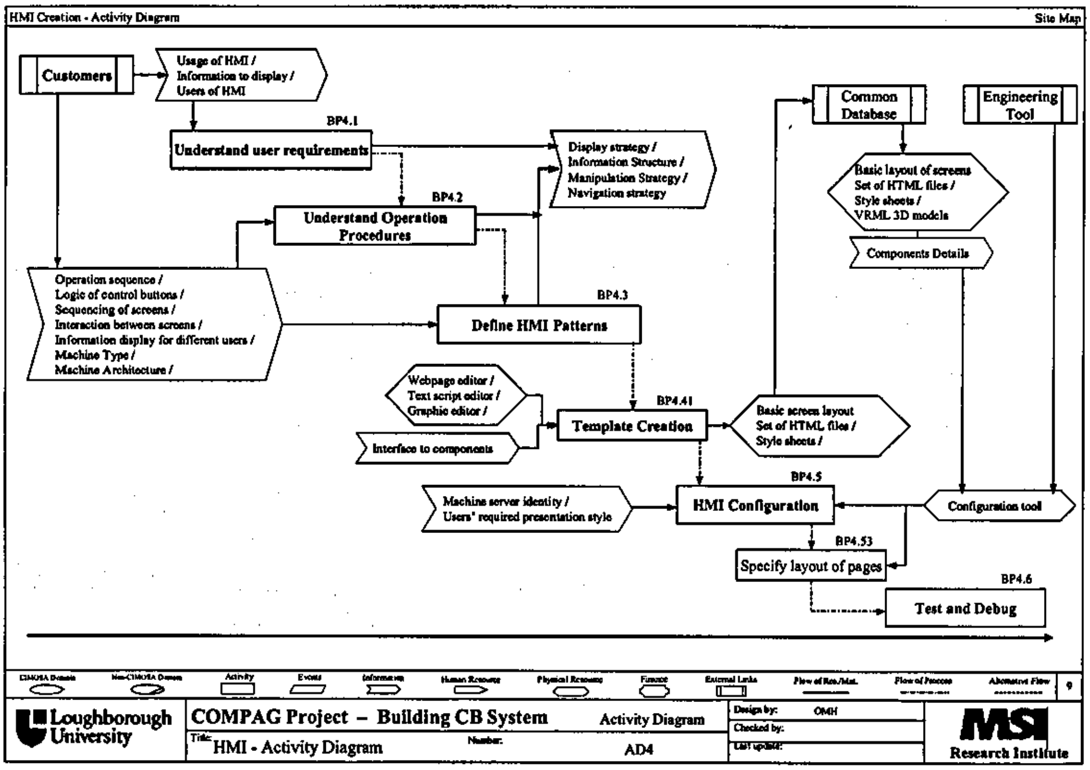


Figure 6.30 Activity diagram for creating HMI

6.3.1 Reasons for using CIMOSA diagrams to represent CB system creation

Using the CIMOSA based diagrams to represent the processes in creating a CB system is extremely useful. Firstly, the models created enable the analysis of the CB system. They provide a clear description of the functions, processes and activities involved in developing a CB system. By describing the processes of how systems are created under this paradigm, a potential user could possibly use the model as a guide to the effort required and skills necessary to build a CB system. The models also present the resources that are required in order to develop CBA systems. CIMOSA based models also can be used to enable a better management of the complexity involved in describing the system. The models simplify the CB system, breaking down the complete system into various modules (i.e. domains) so that each could be examined carefully. Finally, the models could potentially be used to support system design. The models identify the existing functions within the system and their particular capabilities; they could also assist the designer to decide what other functions or services need to be added to the system to meet their individual requirement.

6.4 Summary

The CIMOSA reference architecture and its modelling constructs have been described in this Chapter.

From the modelling point of view, CIMOSA offers a reference architecture and the particular architecture which are suitable for describing the enterprise at generic, partial or particular layers. The models could be derived at different phases of the life cycle, namely, requirement definition, design specification and implementation description. CIMOSA proposed the description of an enterprise from four main views, function, information, resource and organisation. From the function view, the main modelling constructs of CIMOSA are events, domain, domain process, business process, enterprise activity and functional operation. These constructs are the major components used to model the functionality and the behaviour of an enterprise. The modelling constructs are graphically represented in this research using a set of four diagrams, the context diagram, interaction diagram, structure diagram and activity diagram to capture the processes and activities within an enterprise.

The iThink™ modelling software has been used to create dynamic models of the CIMOSA representation in this research. Although iThink™ is not CIMOSA compliant, it has a set of general purpose and simple model building blocks which could be easily adapted to cater for the simulation of CIMOSA-based models. The constructs have been structured to enable a direct mapping from CIMOSA diagrams to iThink™ models, to ensure consistency and enable reuse of functionality for different processes if required.

Finally, the adoption of the CIMOSA modelling approach for representing the creation of a CB system had been discussed. This representation of the engineering processes is useful as it: i) provides support for analysis of the CB system, ii) promotes understanding of the CB system, iii) enables a better management of the complexity involved in describing the system and iv) potentially, can support system design.

Chapter 7

Case Studies

7.1 Introduction

The formulation of an evaluation strategy has been considered earlier in this thesis (Chapter 4). Also discussed, were methods that enabled knowledge to be collected (Chapter 5). A modelling method has also been described to represent processes for the design and build of engine production machinery (Chapter 6). The application of the evaluation approach and modelling method to the CBA as developed in the CCG projects is discussed in this Chapter.

The CBA developed under the CCG projects was implemented and tested at two machines at collaborators' sites. One demonstration was a transfer line machine at Lamb Technicon in United Kingdom and the other one was an assembly machine at Johann A. Krause in Germany. A bench top prototype test rig at Loughborough University, developed by the CCG projects for proof of concept implementation was also used in the evaluation research. The evaluation method was implemented in these three case studies and the results are presented in the following sections.

7.2 Case Study 1: Lamb Technicon

Lamb Technicon specialises in the design and production of dedicated and flexible transfer-type machine tools and integrated manufacturing systems. It is a Tier-1 supplier to Ford Motor Company for transfer line machines. In 2003, Lamb Technicon merged with Cincinnati Machine to form Cincinnati Lamb. The company has several operations around the world, including United States, Canada, United Kingdom and Germany. The project collaborator for the CCG projects was the United Kingdom branch Lamb Technicon UK (referred to as Lamb in the rest of the Chapter).

7.2.1 Understanding Lamb through Interviews and Analysis of Documents

The project work (machine design and implementation and knowledge elicitation) at Lamb Technicon commenced in June 2002 and lasted for a year. Several site visits and interviews were conducted with personnel (i.e. project managers, control engineer and mechanical engineer) from Lamb. The work commenced with site visits; during the visits, personnel from Lamb gave a brief introduction to their plant and general descriptions of the business and their roles and responsibilities. Interviews were arranged. The first interview was a group interviews and subsequently one-to-one interviews were arranged. In total, eight interviews were conducted. The interviewees were highly experienced and were able to provide information on the different aspects of the organisation and the engineering processes. Through these interviews, an overall understanding of the engineering processes was obtained. However, due to limited time and confidentiality issues, certain aspects of the organisation were unable to be covered e.g. financial information and sensitive timing and operational data.

The implementation of the CBA system on Lamb's test machines was concluded with a commissioning test. A control engineer from Lamb conducted the test to assess if the machine implemented using CBA met the industrial requirements.

A detailed description of the machine design and build processes in Lamb is documented in Appendix C. A brief description is presented here. The typical period for the development of one transfer line is about 50-53 weeks but there is pressure to reduce this development time to 42 weeks. Besides designing the engine production machine, Lamb provides simultaneous engineering (SE) services. At Lamb Technicon UK, a lot of the work is sub-contracted, such as mechanical design and the building of parts. Lamb's Marketing and Sales department liases with the customer when approached for a quotation for a new engine production machine. If a contract is awarded, the Project Management Department, which manages projects, takes over the responsibility of coordinating the project. The Control Department is responsible for the control, hydraulic and electrical design of the machine. There is a Commissioning Department which takes charge of commissioning the machine. When the machine has been installed, Lamb will perform a series of test and checks to ensure that the machine has met the customer's specification before taking the machine apart and shipping the machine to the customer site for installation. The customer monitors the development work closely by having meetings and sending their engineers to Lamb. At different stages of the project, the customer's approval is required before Lamb proceeds with the work.

7.2.2 Process Modelling

An explanation of how the CIMOSA modelling framework has been adopted for representing the processes in Cross Hüller (i.e. using MMA) has been given in Chapter 6. The Cross Hüller's diagrams have been used for illustration purposes only since the CBA was not implemented at Cross Hüller and hence did not provide a good test case for comparison. As a CBA machine was implemented at Lamb, it has been used as a case study in the evaluation research.

The overall project domain and the areas of focus as shown in Figure 6.5 and Figure 6.6 were the same for all the machine builders, i.e. Cross Hüller, Lamb Technicon and Krause, since they were collaborators on the same project. However, the two diagrams have been included here as Figure 7.1 and Figure 7.2 for clarity.

The overall context diagram, i.e. a breakdown of the functions within Lamb is shown in the context diagram in Figure 7.3. The CIMOSA domains were classified as: "Marketing and Sales", "Project Management", "Sub-Contracts", "Assembly", "Purchasing" and "Shipment and Installation". An interaction diagram, as shown in Figure 7.4, has been constructed to show the different information and resources exchange within the organisation and also with their customers. For example, in Figure 7.4, the customer makes a request to Marketing and Sales for (i) an SE meeting, (ii) the generation of a proposal and (iii) the financial issues (i.e. a detailed breakdown of costs) involved in the project. In response, Marketing and Sales, will arrange for the SE meeting, put in a proposal and give a presentation of the proposal. When both parties are satisfied with the proposal, the contract is signed. Marketing and Sales will produce a document, known as the "Green Book" for internal circulation. This Green Book contains all the details about the deal: the machine specifications, the financial details, project schedules etc. The Green Book, the order specification and sale schedule are given to the "Project Management".

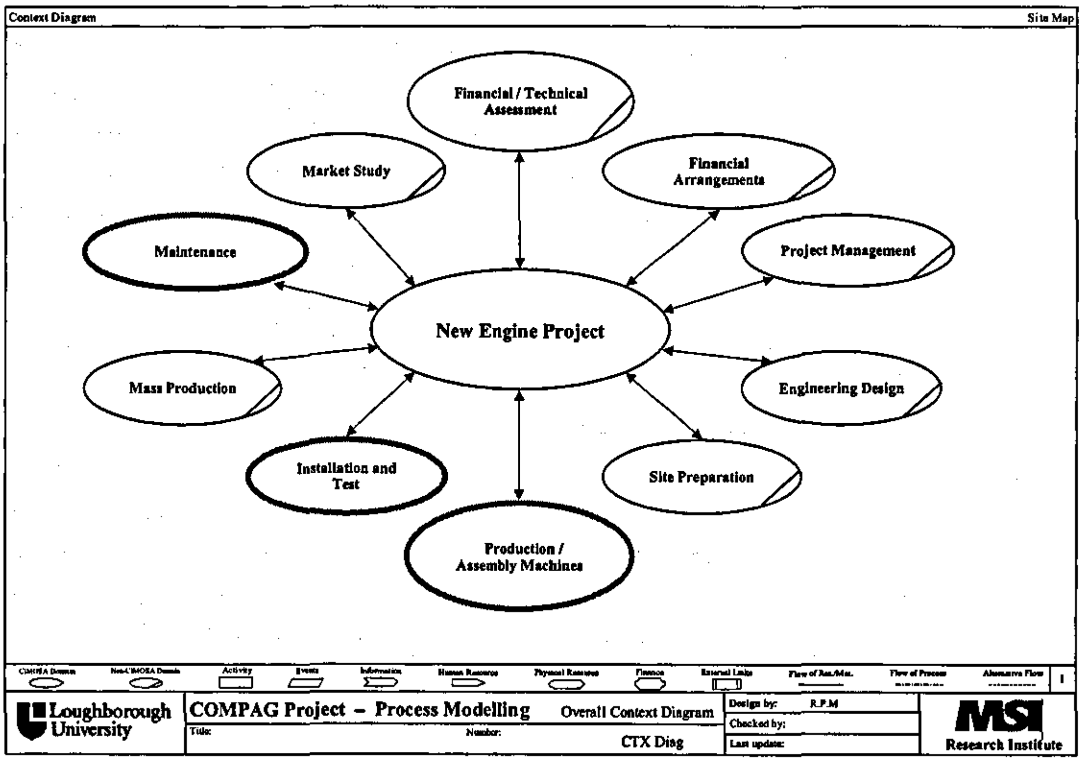


Figure 7.1 Overall context domain for New Engine Project (Source: Monfared 2002 [31])

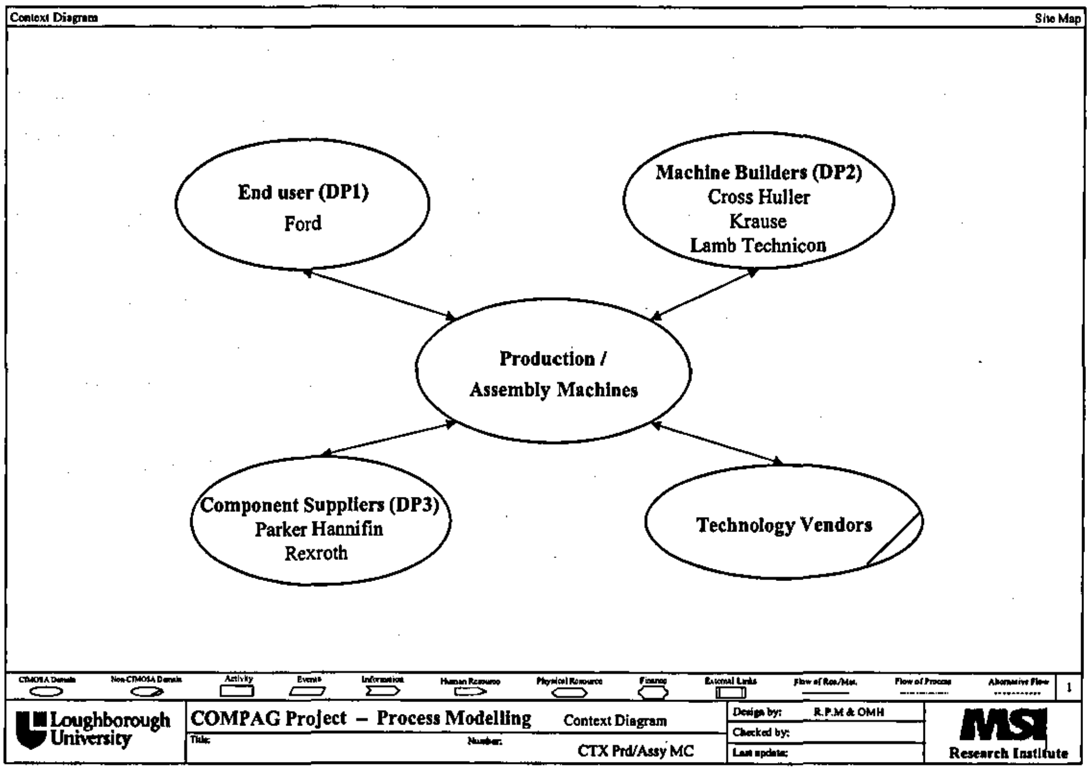


Figure 7.2 Context diagram of Product / Assembly Machines

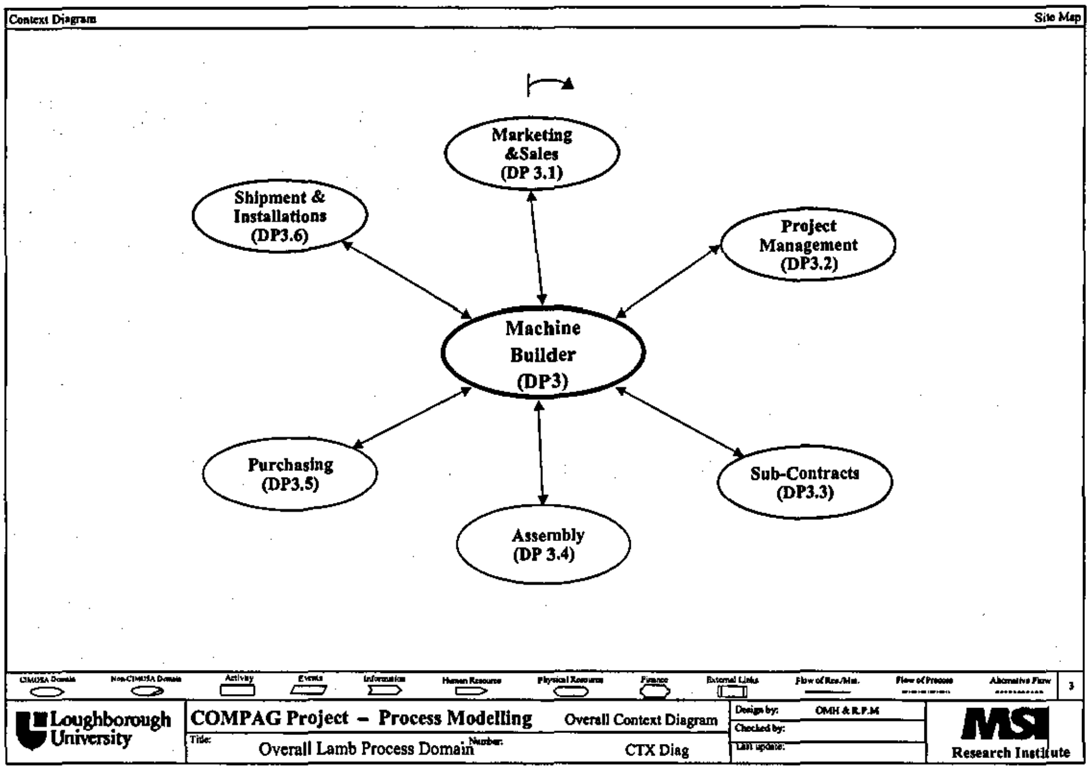


Figure 7.3 The overall context diagram in Lamb Technicon

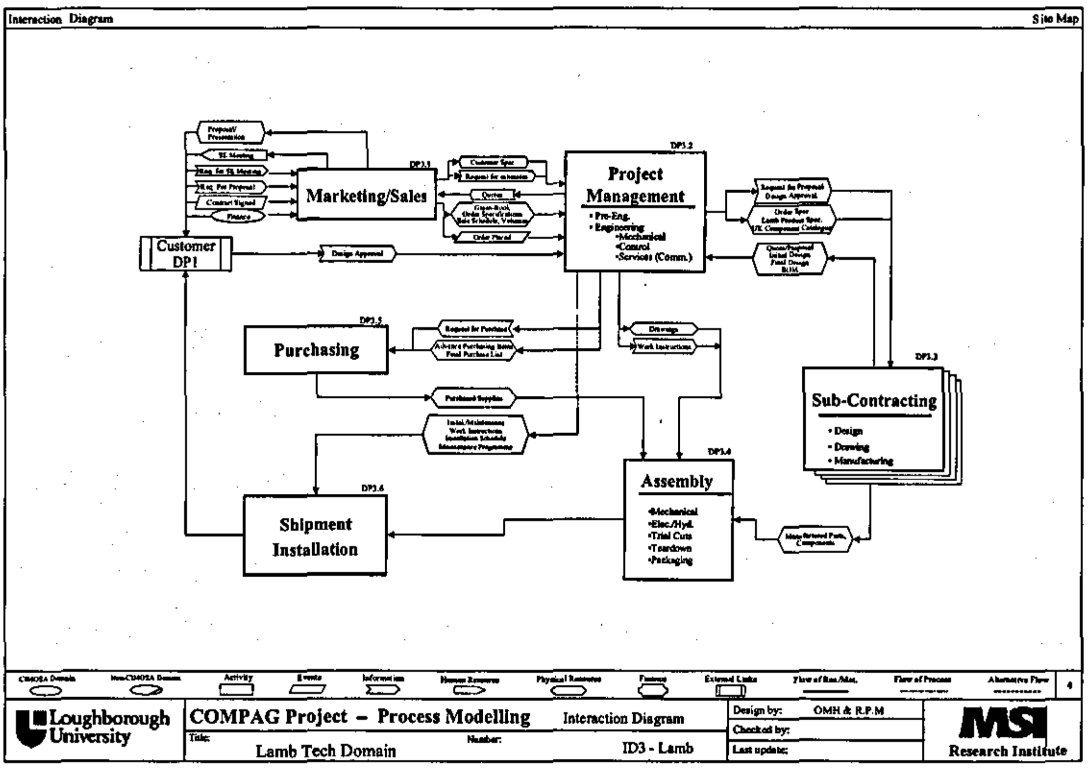


Figure 7.4 Interaction diagram for Lamb

The structural decomposition within Lamb is shown in Figure 7.5. In the diagram the domain processes (DP) are further decomposed into business processes (BP) and enterprise activities (EA). For example, Marketing and Sales (DP3.1) comprises "Attending SE Meeting (EA3.1.1)", "Developing Proposal (BP3.1.1)", "Sale Order (EA3.1.2)", "Finance Issues (EA3.1.3)", "Prepare Greenbook (EA3.1.4)", "Load BOM (i.e. Bill of Materials covering the complete range of parts required for the machine) (EA3.1.5)" and "Lining up initial vendors / suppliers (EA3.1.6)". The activities involved in developing a proposal (i.e. Developing Proposal BP3.1.1) are: i) to prepare an initial proposal (i.e. EA3.1.1.1), ii) to provide a quotation of the project costs and duration (EA3.1.1.2), iii) to prepare the proposal drawings of the machine design (EA3.1.1.3) and iv) finally to give presentation (EA3.1.1.4) of the proposal to the customer.

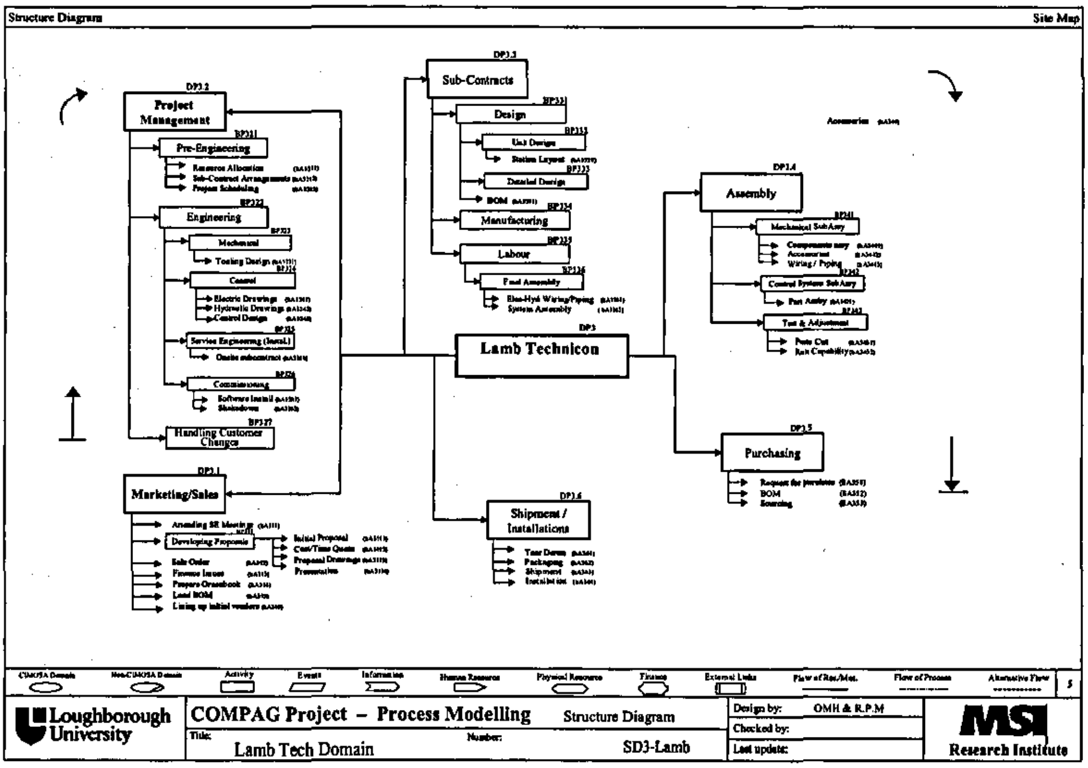


Figure 7.5 Structure diagram for Lamb

At Lamb, the beginning of the Pre-Engineering phase (BP3.2.1) within the Project Management Domain (DP3.2) after the contract has been signed marks the start of the machine design and build project, as shown in the activity diagram in Figure 7.6.

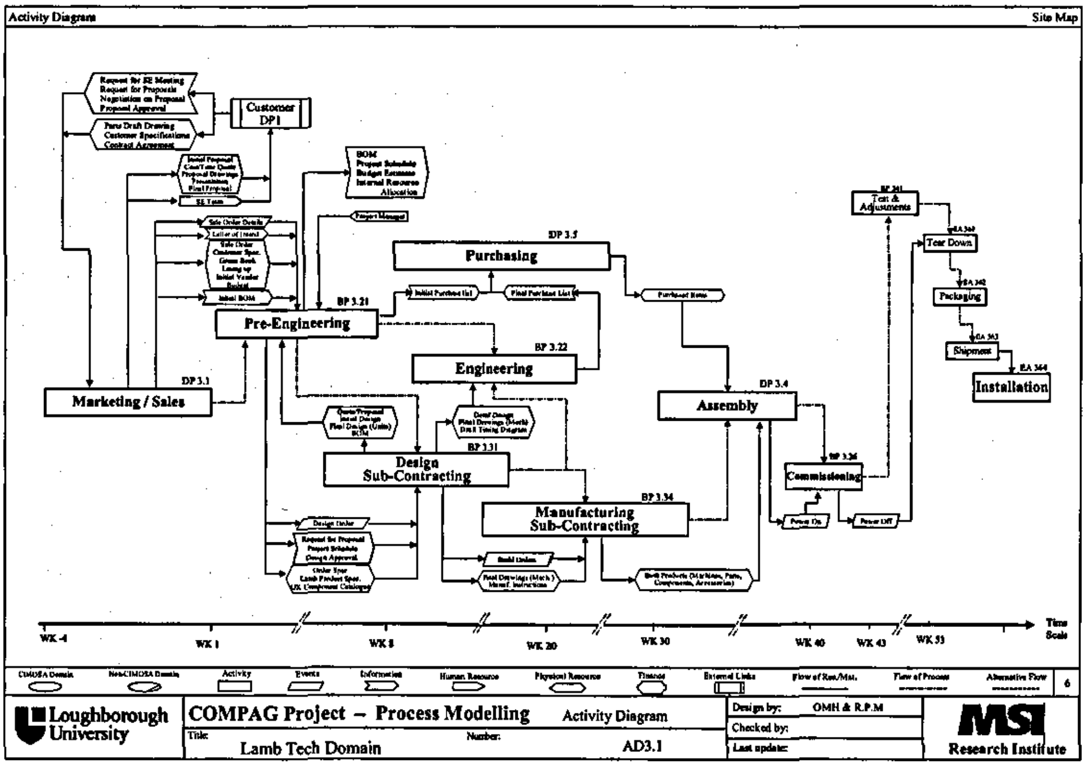


Figure 7.6 Activity diagram (I) for Lamb

In the diagram the time sequence of activities undertaken by Lamb from Marketing and Sales through to Commissioning (BP3.2.6) and finally Installation (DP3.6) are illustrated. The information, resources and events that are specific to each process are indicated. For example, Marketing and Sales has to provide Pre-Engineering with the sales order details, the letter of intent (i.e. letter of agreement), customer specifications, the Green Book, an initial list of preferred vendors, budget details and the initial BOM. The preparation for sub-contracting the design work takes place during pre-engineering. A request for proposals is sent to the sub-contractors along with the project schedule. At the same time, Lamb provides the customer specification, Lamb's product specification and a British component catalogue to the sub-contractors. The Engineering process (BP3.2.2) usually starts midway through the "sub-contracted" design process (BP3.3.1). The sub-contractors provide the detail design, final mechanical drawing and the draft timing diagram for Engineering. The manufacturing process, which is also sub-contracted (Manufacturing Sub-Contracting BP 3.3.4) starts near the end of the design process. Building orders have to be issued and final mechanical drawings and manufacturing instructions are given to the manufacturers. In return the manufacturers provide the build products (i.e. machines, parts, components, accessories). Assembly of the complete machine (DP3.4) begins at Lamb when the external manufacturing work has been completed. Items that have been purchased are required at this stage. Following the Assembly process the machine undergoes extensive Commissioning activities (also referred to as "Power On" within the company). Tests and adjustments are carried out after Commissioning and when all the testing and adjustments have been completed, the machines are taken apart, packed and shipped to the customer's shopfloor for installation.

The second activity diagram in Figure 7.7 focuses on the time sequences and the interaction between the engineering, assembly and commissioning activities. It provides a more "local" detailed view of the activities. For example, prior to the Final Assembly process, Mechanical sub-assembly and Control sub-assembly have to be completed. Similarly, Control software installation is followed by "Shakedown" (i.e. running the machine without any parts), parts cut and a run capability check before the machine is ready for shipment. The Customer has to provide the raw parts for parts cutting and has to give their approval to the results of the parts cut and the capability check activities before the machine can be shipped.

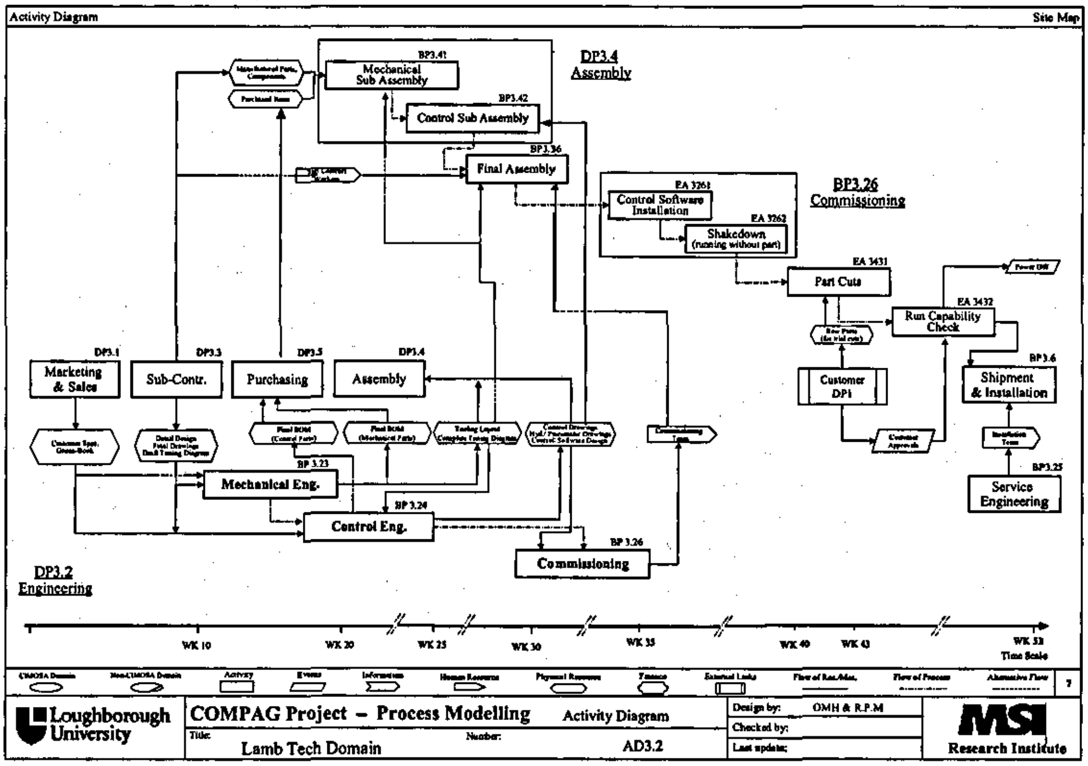


Figure 7.7 Activity diagram (2) for Lamb

The main difference between Cross Hüller and Lamb is in their sub-contracting activities. Cross Hüller has its own mechanical designing team to do the machine design whereas Lamb sub-contract not only the mechanical designing, but also the manufacturing work. Thus, in the activity diagram for Lamb (Figure 7.6), a lot of sub-contracting activities are shown, such as issuing design orders, requests for proposals, requests for project schedules, issuing design orders and sub-contracting labour. It is apparent in this case that the adoption of the CBA by Lamb would also have a major impact on their sub-contractors. This will be discussed in Chapter 8.

7.2.3 Scenario Testing

Scenario testing has been used to assess the impact of implementing the CBA on a standalone transfer line machine at Lamb (illustrated in Figure 7.9 and as VRML model in Figure 7.10). Transfer line machines are commonly used in the automotive manufacturing sector for the machining of parts such as engine blocks, transmission cases or cylinder heads in high volumes [176]. In a transfer line machining system, several machining stations (commonly known as wingbases) and fixture mechanisms are lined up along a central transfer line mechanism, as shown in Figure 7.8.

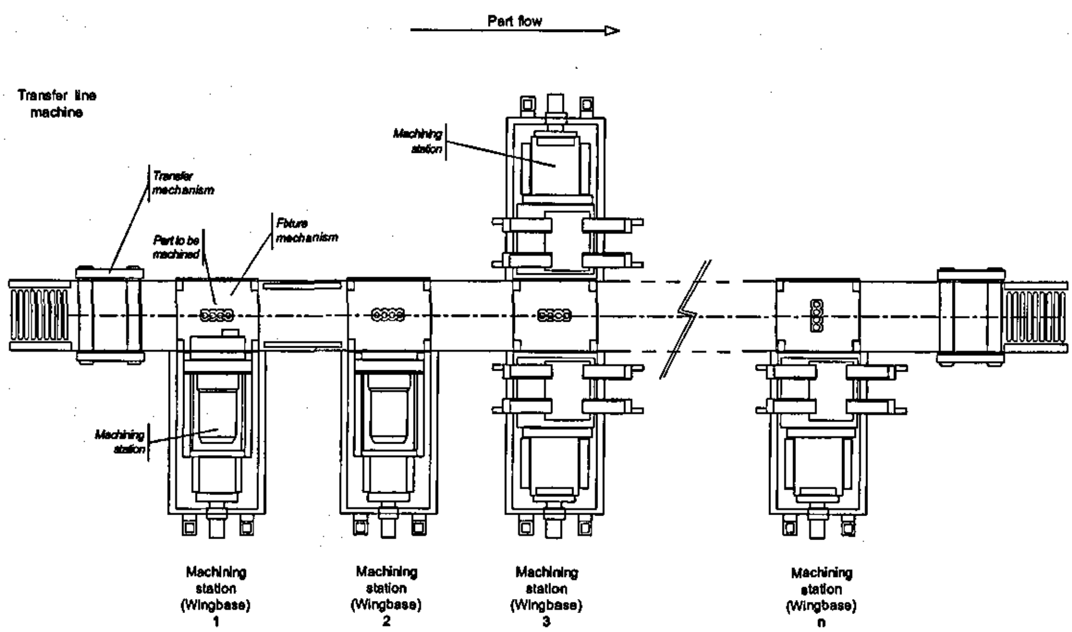


Figure 7.8 Schematic of a transfer line machine (Lee 2004 [71])

The test machine at Lamb consists of a transfer mechanism (a transfer bar) and one cylinder-head machining station (wingbase consisting of a clamp and a cross slide).

Parts are transferred sequentially via the transfer mechanism from one machining station into the next. When a part arrives at the machining station, a hydraulic fixture is lowered to clamp the part. At the machining station, a horizontal cross slide is used to position a drill spindle and servo drive to perform drilling operation on the part. The machine is used by Lamb to evaluate new technology prior to adoption.

The control system of the test machine was implemented according to Lamb's specifications. Using the CBA, a total of 10 components, a VRML model and HMI were implemented. The demonstration of the capability was part of Ford Motor Company's global commonality meeting attended by Ford and all of their Tier 1 and Tier 2 suppliers.

Figure 7.9 Lamb Test Machine

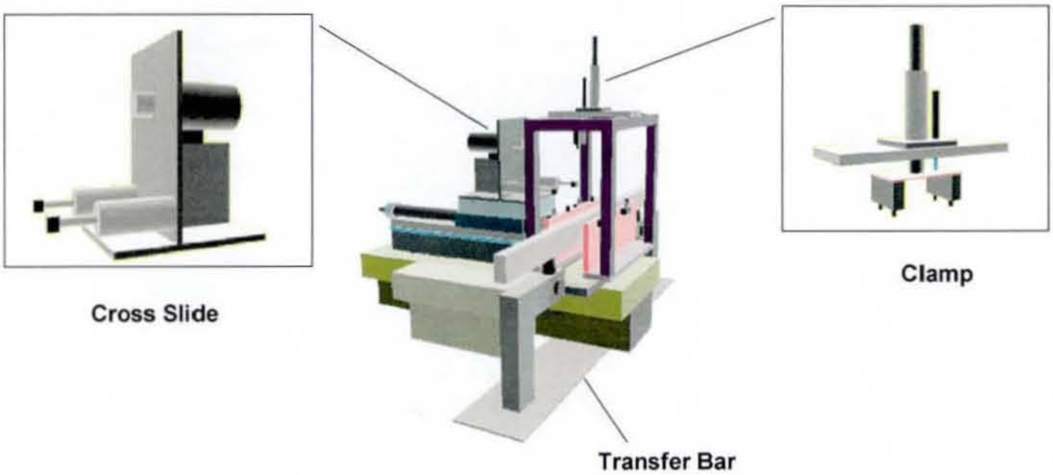


Figure 7.10 VRML model of Lamb Test Machine showing the different mechanism in the machine

7.2.3.1 System Design

Setting: *A new machine is to be designed.*

Question: *How long will the new CBA system take to design the machine?*

Metric: *Time*

A test was conducted to determine the time taken for the commissioning of the test machine at Lamb.

Based on the data from Lamb, the electrical design of the test machine takes 80 hours, control design 45 hours and commissioning 40 hours. Using CBA, ten individual components were implemented. These can be be categorised into eight different types, as shown in Table 7.1.

Subsystem	Component	Type
Transfer Subsystem	Monitor console	*1*
	Transfer raise/lower	*2*
	Transfer advance/return	
	Hydraulic service	*3*
	Air/lubrication services	*4*
Fixture subsystem	Clamp	*5*
	Part seated sensors	*6*
Wingbase Subsystem	Monitor console	
	Head/Cross slide	*7*
	Servo drive	*8*

Table 7.1 System decomposition for Lamb test machine (adapted from Lee 2004 [71])

Based on the estimation in Chapter 6 on developing a component using the CBA (section 6.3), the total number of days that is required to develope the eight types of components in Lamb are as followed:

Total component development time = Duration of 1st component + (8-1) (Duration of subsequent component)

= 11days + 7(6.5 days)

= 11 days + 45.5days

= 56.5 days (11.3 weeks)

Assuming that there are five working days per week, implementing the eight types of components from scratch would take 56.5 days or 11.3 weeks. If, however, the CBA products are already available commercially off the shelf (COTS), the CCG project implementation will take 40 hours[14] for electrical design, 24 hours for control design and 24 hours for commissioning. Assuming an eight hours working day, Lamb would currently require a total of about 20 days for the control system implementation and commissioning, while the CBA would take 11 working days. The design and implementation of HMI have not been considered.

The results of time taken under current practice, developing the components using CBA and the use of COTS are as tabulated in Table 7.2 and illustrated graphically in Figure 7.11. The initial time required to develop the components is very long compared to Lamb's current practice. However, it is important to note that Lamb's design time is a result of reusing previous control design with necessary modification. It is not the actual time required to develop the control system from scratch. Hence, Lamb's figure is estimated based on reuse. For the CBA, although the initial time investment is very high, however, it can be seen that when these components are reused, the time required for control design is reduced drastically, as in the case of CBA-COTS. Moreover, reuse of components is also associated with assured quality, improved quality and risk mitigation (section 2.3.3). These savings contribute to the overall time reduction which will be discussed in 7.2.4 of this Chapter.

The mechanical design of the test machine has not been considered in this scenario because the CCG projects' main focus has been on implementing the control system. However the vision is that 3D representation supplied as part of each component could become an essential part of the mechanical design process and possibly result in similar time and cost savings.

[14] There are currently no specific tools developed under the CCG projects to facilitate electrical design. The figure has been estimated based on the assumptions that by using CBA, low level implementation of electrical design are already encapsulated within the component. A user will be able to use the component directly using the standard electrical interfaces.

	Lamb	CBA (from scratch)	CBA (COTS)	% Difference between Lamb & CBA(COTS) *(based on the hours)*
Electrical Design	80 hours (10 days)	40 Hours[15] (5 days)	40 hours (5 days)	50%
Control Design	45 hours (5.1 days)	56.5 days	24 hours (3 days)	46%
Commissioning	40 hours (5 days)	24 Hours [16] (3 days)	24 hours (3 days)	40 %
Total	20.1 days	64.5 days	11 days	45 %

Table 7.2 The number of hours and days that are required for machine control system implementation and commissioning under the current practice, CBA and CBA(COTS)

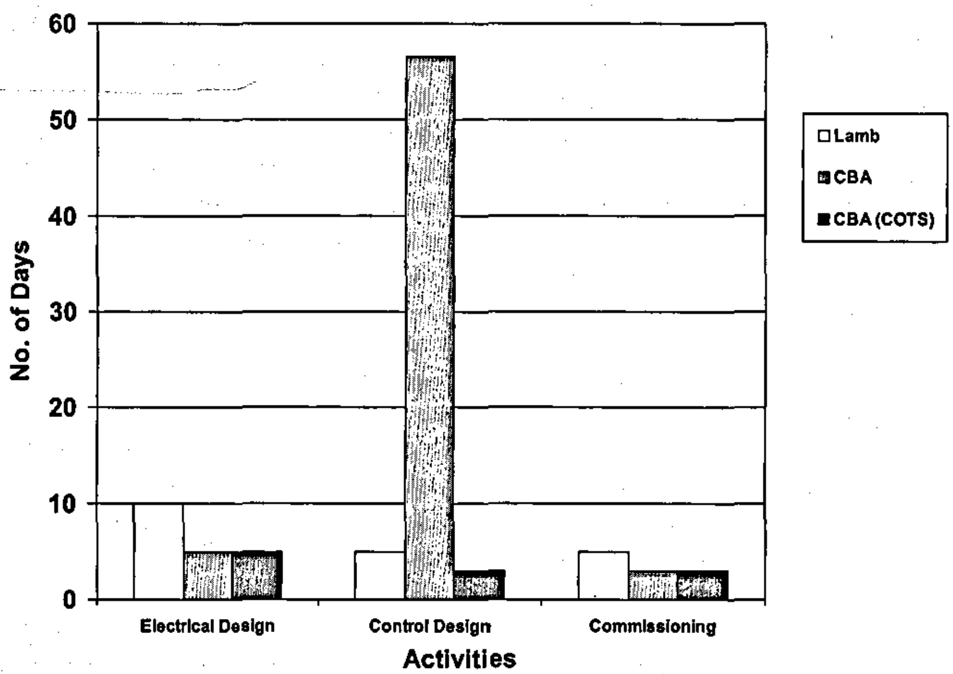


Figure 7.11 Graphical comparions of the current practice and CBA

[15] The time required is the same as in the case for COTS because this is the time required for the electrical design of the complete system.

[16] The time required for commissioning will be the same as in the case of COTS because individual behaviour of the components had already been tested during component development. The commissioning time in this case is the time required to commission the system.

7.2.3.2 System Manipulation

Setting: *The system is processing a part and an unexpected error occurs. The on site user must now manually manipulate the system to bring it back to normal operation.*
Question: *What needs to be done to bring the system back to normal operation in the new CBA system?*
Metric: *Number of steps*

When a part is suddenly removed from the machine, an error message is raised on the human machine interface (HMI). The process of bringing the machine back to normal operation under current practice is as followed:

1. Change the operation mode to manual mode.
2. Check the error message on the HMI.
3. Verify the error.
4. If the error cannot be verified, consult the maintenance engineer.
5. If error can be verified, return the machine to initial position and change the operation mode to automatic mode and restart the machine.

This is represented graphically in Figure 7.12 using the CIMOSA activity diagram.

There is no difference in the steps taken to recover the machine under the current system and the CBA system. Six steps are required to recover the machine in both cases. It should be noted that these are the steps taken by an operator to recover the machine into a working state; the operator is only concerned with using the HMI to control the machine, anything beyond that has to be referred to the maintenance engineers. Therefore, in this scenario, it shows that the operation of the machine under current practice and under CBA makes no difference to an operator. In fact, it was not expected that there should be any differences because the steps to bring a machine to normal operation is a standard procedure. The test on machine manipulation from the point of view of the maintenance engineer and other engineers were not conducted.

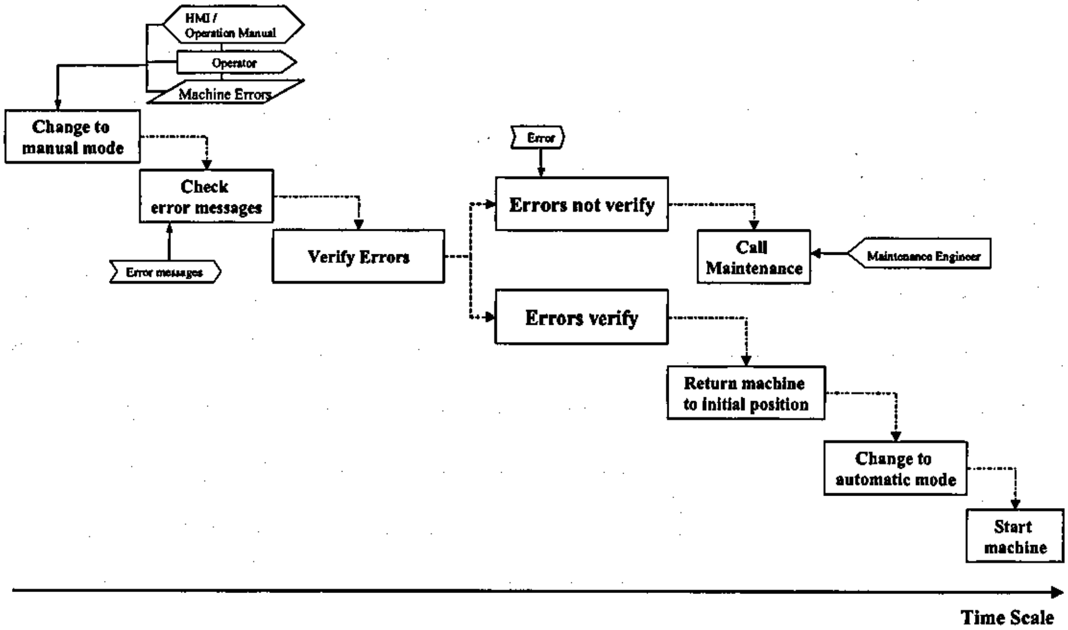


Figure 7.12 CIMOSA activity diagram describing the steps taken to bring a machine back to normal operation when a part is removed from the machine unexpectedly.

7.2.3.3 Modification of Design

Setting: *Assuming that a change needs to be implemented on the machine, what are the impact on the mechanical, control and electrical design and implementation processes?*
Question: *How does the implementation of CBA facilitate the changes in design?*
Metric: *Time*

An analysis of one of Lamb's development projects was undertaken to determine possible impact the CBA would have on the mechanical, control and electrical design processes. From the discussion with Lamb's mechanical engineers, it was found that majority of the changes in a project affect mechanical design. When the mechanical design changes the electrical wiring also has to be changed. Mechanical changes tend to have less impact on the control design. According to Lamb's control engineers, there is relatively less impact on the control design because the control structure, the control system (i.e. usually Programmable Logic Control (PLC)) and the HMI have been unaffected by the typical mechanical changes experienced in the past.

From the discussion in Chapter 4 no CBA tools have been specifically developed for mechanical and electrical design. The Process Definition Environment (PDE) in the CCG project has the facilities for the configuration of the machine control logic.

Although the 3D VRML machine models could potentially be used for mechanical design, it has not been developed for this purpose and currently, it does not include the level of detail that can be provided by Computer Aided Design (CAD) drawings.

Based on this observation, it follows that the implementation of the CBA would not currently have much impact on the mechanical and electrical design but would be able to facilitate control design. As shown in Section 7.2.3.1, control design using CBA is faster than the current approach.

7.2.3.4 Process of Translation

Setting: *The machine builder has the specifications to design a new machine. The various engineers (e.g. mechanical, electrical, control, fluids) involved are to design the various parts of the machine based on these specifications.*
Question: *How would the data be available and be shared amongst the engineers when the CBA has been implemented?*
Metric: *Number of times in which translation is required.*

The process of machine design supported by Lamb, from production specification to the creation of a machine can be found in section 2.2.2, section 7.2.2 and in Appendix C. In this scenario, the current process of machine logic creation is compared with that supported by the CBA (Section 4.2.3).

A simplified CIMOSA activity diagram that describes the process under current practice is illustrated in Figure 7.13 and the equivalent of CBA processes (i.e. the process for machine logic creation using the PDE) is given in Figure 7.14.

It can be seen that under current practice (Figure 7.13), two different tools (e.g. MS Excel and proprietary software) have to be used for creating the timing diagram and the ladder logic and the documents are translated from one form to another (i.e. (i) product specification from words to Gantt chart style timing diagram and (ii) timing diagram). In the CBA case (Figure 7.14), the process of creating a timing diagram is done within the PDE (section 4.2.3, Figure 4.4), and subsequently, interlocks can be added to the timing diagrams to create the machine logic. Therefore, it can be seen that it takes four processes before the machine logic is created under current practice (Figure 7.13), but under the CBA using the PDE, it takes three processes. This is due to the fact that all the work is done in one tool and no translation is required. This explains why CBA is faster and less error prone than the current practice.

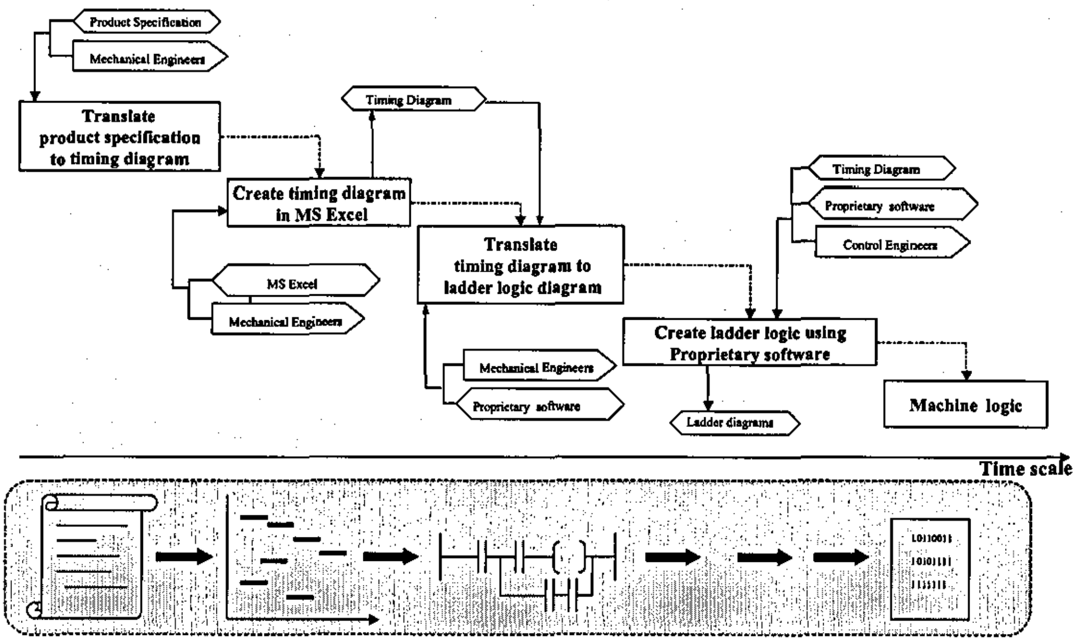


Figure 7.13 A CIMOSA activity diagram is used to describe the process of machine logic creation under current practice; the lower half of the figure is a graphical illustration of the different formats of documents that has been translated at different stages.

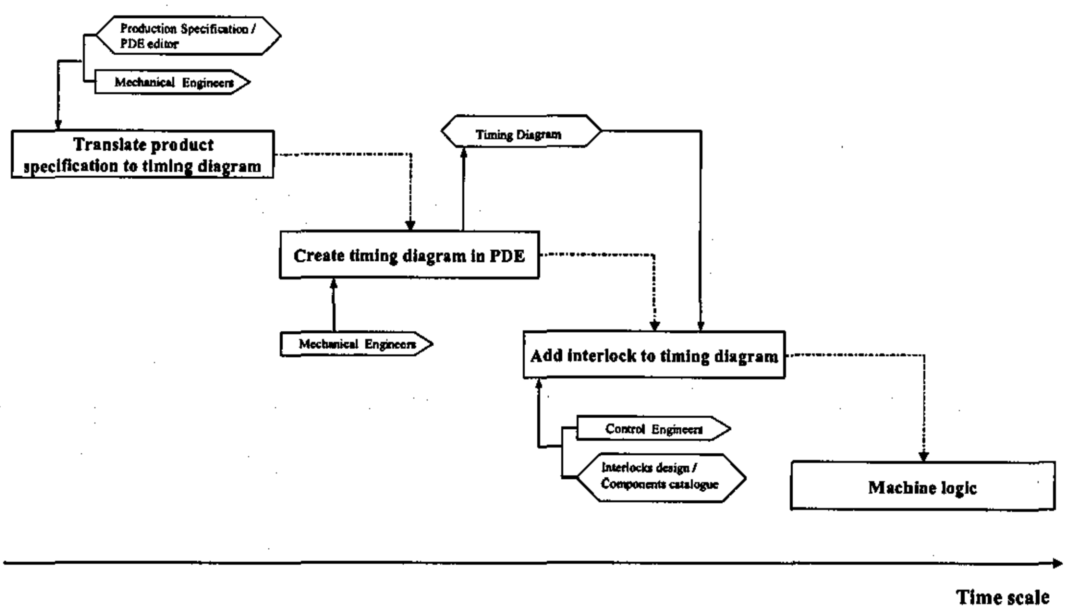


Figure 7.14 Description of the process of machine logic creation using the CBA, in the Process Definition Environment (PDE) using a CIMOSA activity diagram.

7.2.4 Dynamic Modelling

The static models created for Lamb have been used as the foundation for dynamic modelling using the iThink™ modelling tool (section 6.2.6).

The dynamic model consists of four main views: the process, information and resources, conditions and durations (see section 6.2.6) For example the DP "Sales & Marketing DP3.1" represented in Figure 7.15, each of these views are represented within the sector frame in iThink ™ at the model construction layer (see section 6.2.6) and are also automatically generated as the process frame at the interface level as shown in Figure 7.15. This facilitates navigation to the associated sector frame and the stock/flow structure at the model construction layer. Figure 7.16 shows an example of the model created for Lamb, based on the DP "Marketing and Sales DP3.1'. The full set of Lamb's dynamic models can be found in Appendix C. As mentioned in Chapter 6, iThink™ has a set of constructs which can be used in a compatible way to CIMOSA and enable the CIMOSA diagrams to be translated into dynamic model in iThink™ (see section 6.2.6, Table 6.1). For example from the structure diagram in Figure 7.5, the EA "Attending SE Meetings EA3.1.1" has been adapted as "process view" in iThink™, as shown in Figure 7.16a. The different information and resources required, i.e. "request for SE" and "Parts draft drawing", for this activity are illustrated in Figure 7.16b. The time taken to complete the EA, starting and ending conditions of the EA are adapted as "Duration" and "Conditions" view in the iThink™ model in Figure 7.16c.

Based on the information determined within the Lamb case study a number of project milestones were identified: "Sale Order EA3.1.2", "Pre-Engineering BP3.1.2", "Mechanical Sub-Assembly BP3.4.1", "Commissioning BP3.2.6", "Test and Adjustment BP3.4.3", "Tear Down EA3.6.1" and "Shipment EA3.6.3". According to company data, the assembly of the machine usually starts in week 29; the commissioning in week 39; test and adjustment in week 43; tear down and packing in week 48 and shipment in week 55. The AS-IS and TO-BE project timelines were generated from the iThink™ simulation of the i) Lamb processes, and ii) changes to the Lamb process that the CBA would impose (Table 7.3). Based on the scenario testing on system design (Section 7.2.3.1), the changes that CBA would impose to the Lamb process are time savings of: i) 50 % in electrical design, ii) 46% in control design and iii) 40% in commissioning. The simulated results are illustrated in Figure 7.17 and

Figure 7.18. In both the AS-IS and the TO-BE model, the assembly of the machine begins in Week 23 of the project timeline. However, commissioning is able to start earlier in Week 31.50 in the TO-BE model while it starts later (Week 33) in the AS-IS model. The CBA commissioning was completed within two weeks and testing and adjustment was able to commence in Week 33.50. As a result, in the TO-BE model, the machine can be stripped down, packed and ready for shipment by week 48.60, but in the AS-IS case, shipment is only ready in Week 52. This shortening of the project timeline is due to savings in control design time, electrical design time and commissioning time after CBA has been implemented (Section 7.2.3.1).

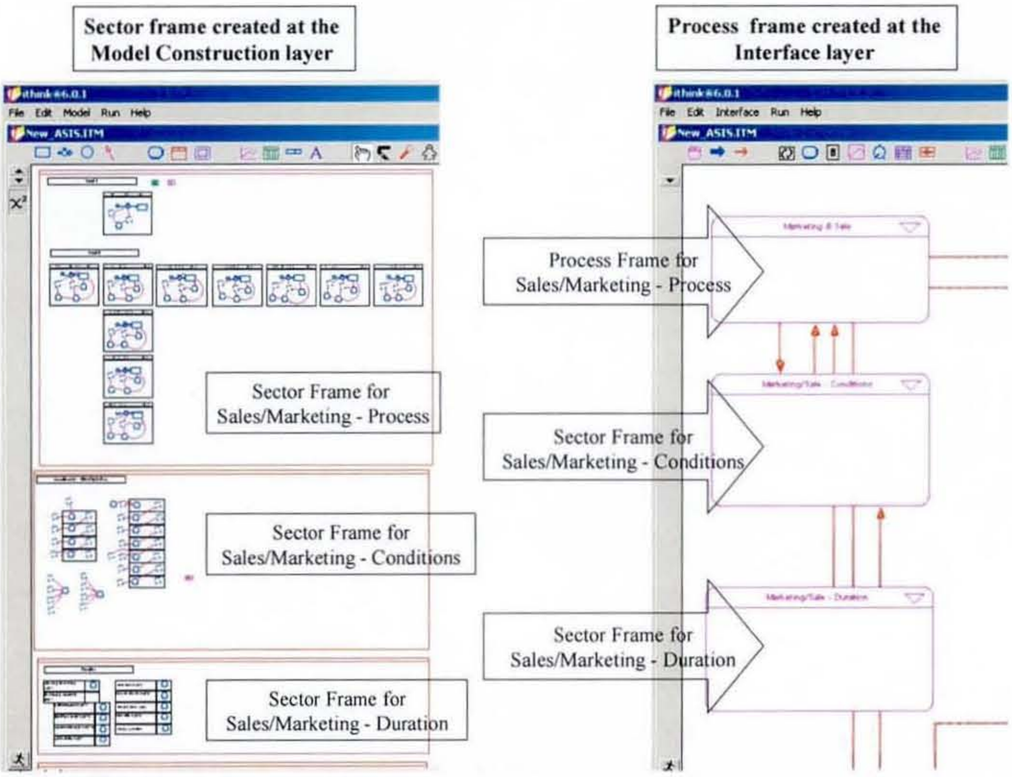


Figure 7.15 The sector frame created at the model construction level and the associated process frame at the interface level in the iThink™ model

Marketing and Sales DP3.1 - Process

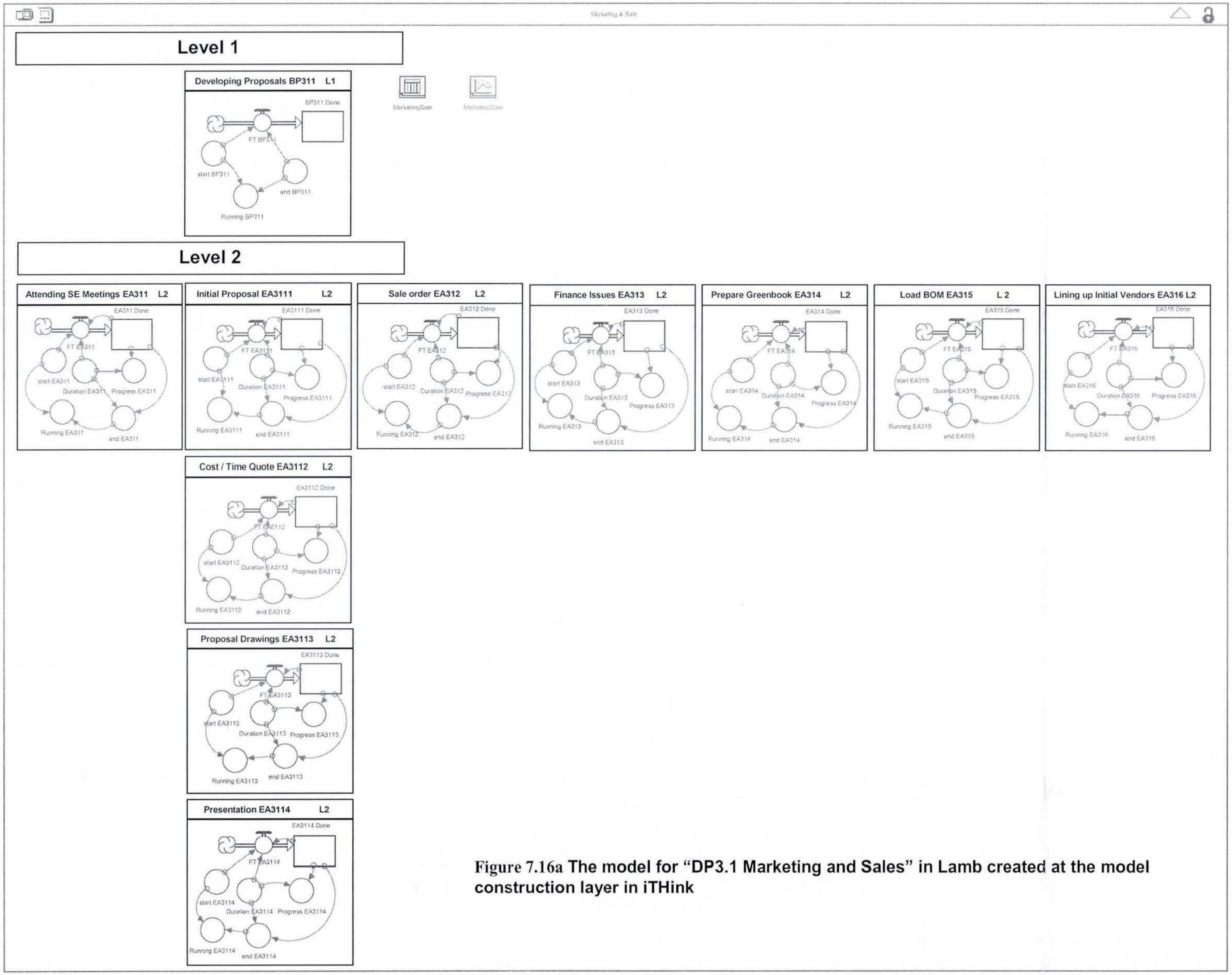


Figure 7.16a The model for "DP3.1 Marketing and Sales" in Lamb created at the model construction layer in iTHink

Marketing & Sales DP3.1 – Information & Resources

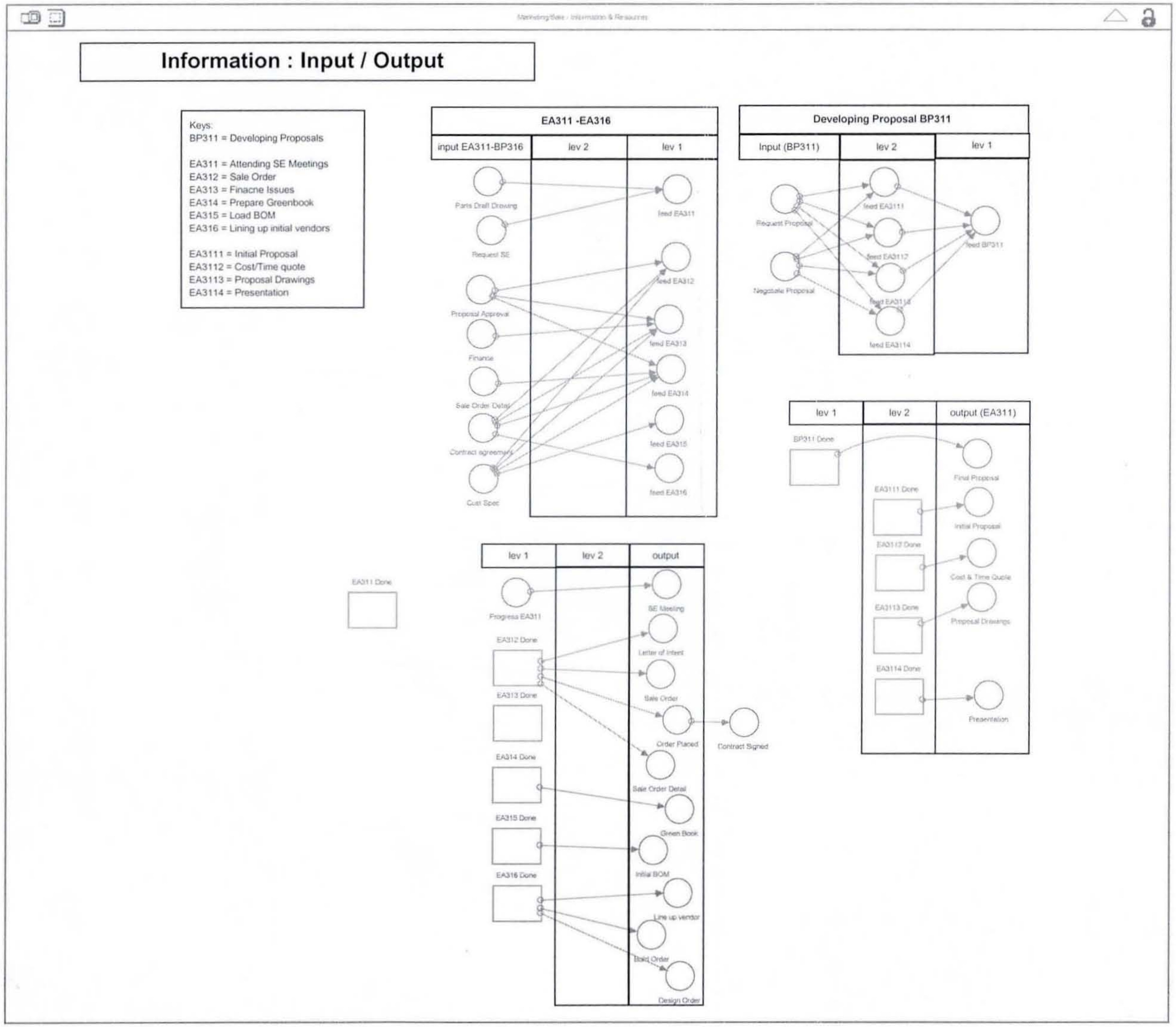
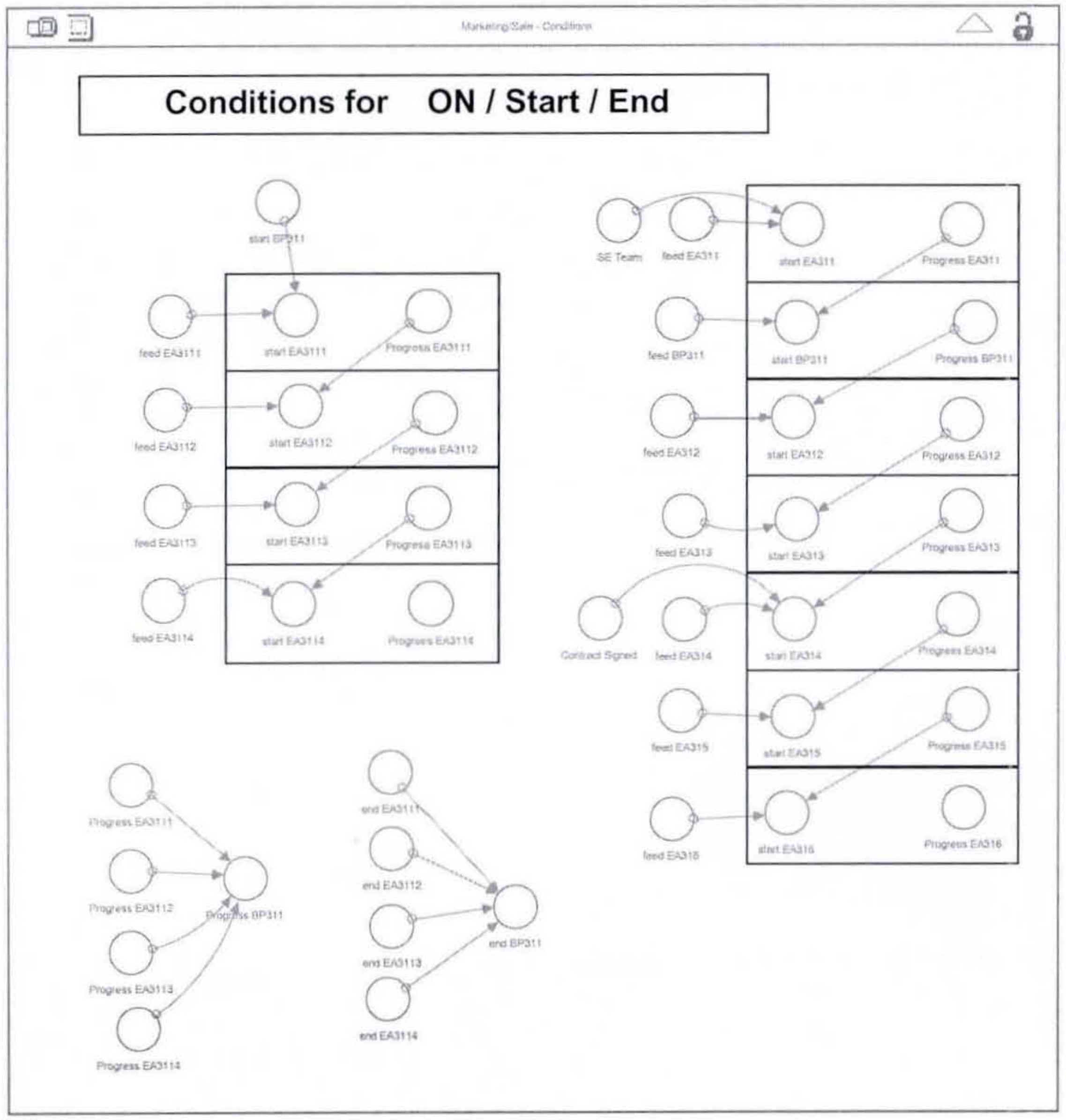


Figure 7.16b The model for "DP3.1 Marketing and Sales" in Lamb created at the model construction layer in iThink

Marketing & Sales DP3.1 – Condition / Duration



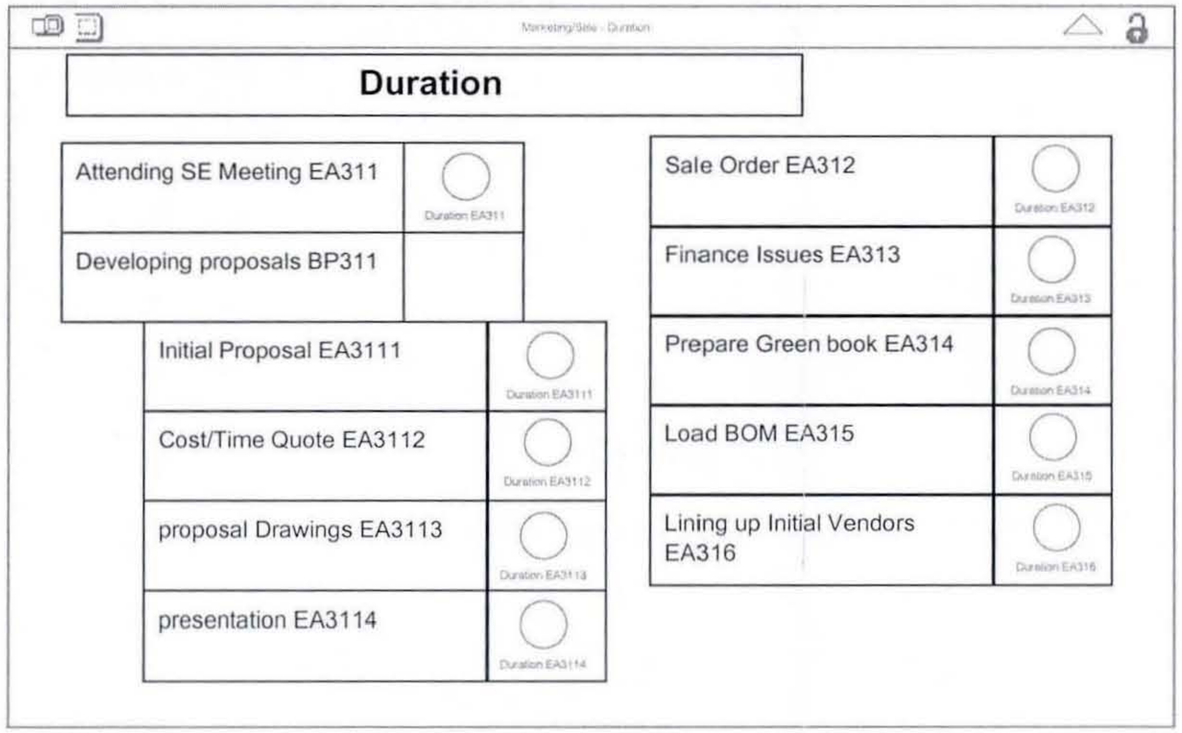


Figure 7.16c The model for "DP3.1 Marketing and Sales" in Lamb created at the model construction layer in iThink

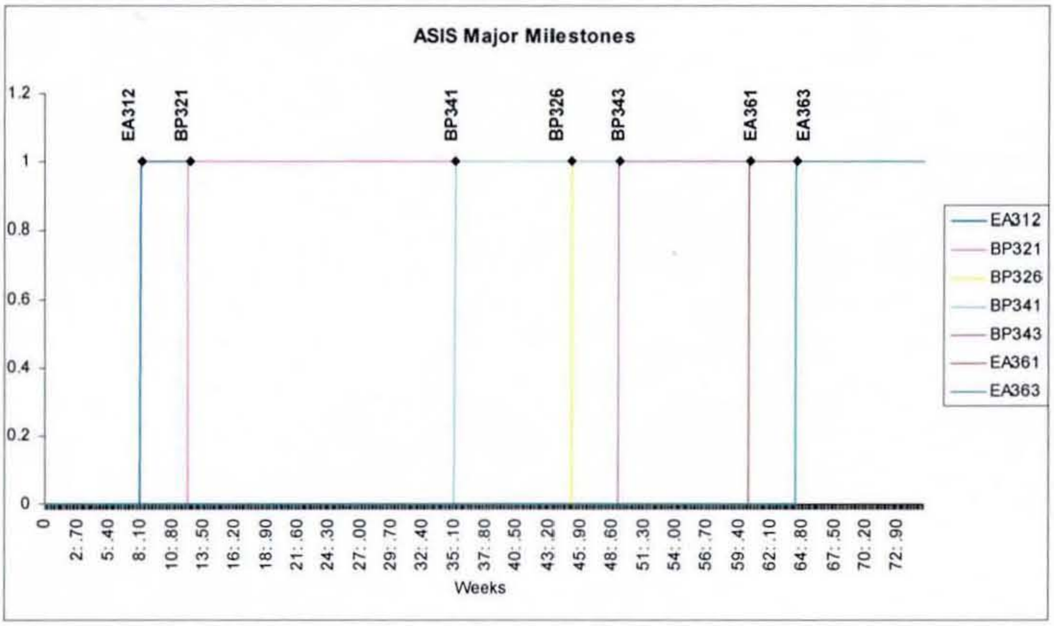


Figure 7.17 Graphs showing when the beginning of major milestone processes and activities in the AS-IS model

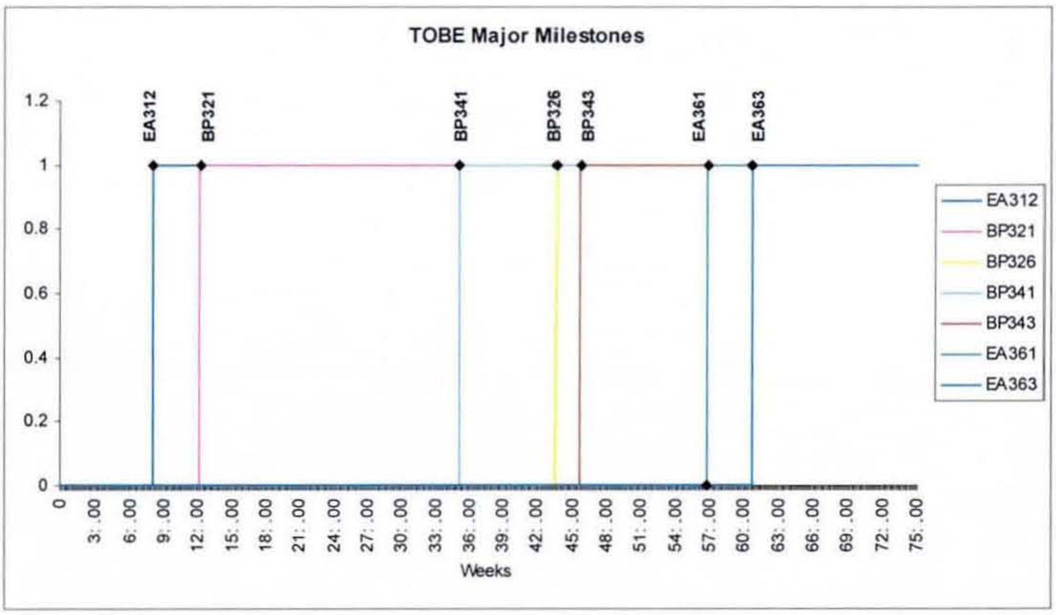


Figure 7.18 The beginning of the major milestone processes and activities in the TO-BE model

	AS-IS Model (Week)	TO-BE Model (Week)
Milestones		
Pre-Engineering (BP321)	1	1
Assembly of machine (BP341)	23	23
Commissioning (BP326)	33	31.50
Test & adjustment (BP343)	37	33.50
Tear down (EA361)	48	44.60
Shipment (EA363)	52	48.60

Table 7.3 Tabulation of the actual project time line and timeline generated from the AS-IS model and the To-BE model

7.3 Case Study 2: Johann A. Krause GmbH

Johann A. Krause GmbH specialises in the design and manufacture of assembly systems for the automotive industry. The company is a major Tier-1 supplier to Ford. Krause, a division of the Thyssen-Krupp Technologies Group, employs more than 2,300 people worldwide, with annual revenue of 400 million Euros. The company's headquarter is in Bremen, Germany.

Krause is the main supplier of assembly machines as opposed to transfer line machines. Assembly machines are used to assemble components of cylinder heads i.e. valves, pistons. Similar to Lamb, Krause has commissioned a purpose built test-rig that is used to enable the issues associated with novel technology to be assessed prior to adoption.

7.3.1 Understanding Krause through Interviews and Analysis of Documents

The Krause test-rig was re-engineered using the CBA at Krause's headquarter in Bremen, Germany. The project work (implementation and knowledge elicitation) lasted for a year. However, due to the fact that engineers were implementing machines worldwide, few interviews could be conducted with relevant Krause employees. Two of the interviews were one-to-one interviews with a project manager. Further information was collected in an ad-hoc manner, via informal conversations with the operation and project engineers on site.

Detailed description of the machine design and build processes employed at Krause is given in Appendix D. A summary of the processes is given in this Chapter. As with the case for Lamb, the project time-line for designing and building an assembly machine is about 50-52 weeks. Krause provides SE services for customers. The sales department in the company works with the project department in developing the project proposal. The sales engineers present a rough concept of how the production line would operate based on previous machine designs to the end user clients.

The project starts when the proposal has been accepted. Each project is allocated to a team of engineers. The human resources for the project are allocated by the manager of each department. Most of the work, i.e. design and build, are undertaken "in-house" in contrast to Lamb; sub-contracting is only carried out if Krause does not have the capability or the capacity for the work.

7.3.2 Process modelling

Process modelling of Krause's work processes was restricted because of: i) difficulty of obtaining time with engineers, ii) language issues, iii) reluctance to discuss any details of working practices, and iv) the limited time and resources that were available. A general model was constructed and thus although it is recognised that the model is incomplete, valuable time and cost estimations could still be obtained.

As in the case for Lamb, a set of CIMOSA based diagrams have been created for Krause. Five domain processes (DP) had been identified as shown in Figure 7.19, which are "Sales and Project", "Production Design", "Purchasing and Finance", "Manufacturing and Assembly" and "Shipment and Installation". An interaction diagram (Figure 7.20) has been used to capture the interaction both internally within Krause and externally with its customers. For example, the customer makes requests to "Sales and Project" for a SE meeting and proposal generation. "Sales and Project" will arrange the SE meeting and present their proposal.

The decomposition of the processes within Krause is represented in a structure diagram in Figure 7.21. The DP, "Project Management" comprises the Administration (BP4.2.1) and Engineering (BP4.2.9) business processes. Further decomposition illustrates that Administration consists of "Assign Project Team (EA4.2.1.1)", "Resource Allocation (EA4.2.1.2)", "Budget and Time Estimate (EA4.2.1.3)", "Project scheduling (EA4.2.1.4)", "Project presentation (EA4215)" and "Handling Customer Changes (BP4211)".

The activity diagrams in Figure 7.22 and Figure 7.23 show the time sequence of the project. In Figure 7.22, the sequence of processes starts from Sales and Project through to installation. The administration process provides information such as the bill of materials (BOM), project schedule, budget estimate and internal resource allocation to the Engineering process. At the same time, an initial purchase list will be given to Purchasing. Manufacturing and assembly begins near the end of the Engineering process. Detailed design, final mechanical drawings, draft timing diagrams and prototype machines are some of the resources that are required within this process. Purchasing will also hand the purchased items to manufacturing and assembly. The outputs from the process are built products such as machines, parts, components and accessories. Commissioning (also known as "Power On") begins after Manufacturing and assembly. Testing and adjustment of the machines take place during the commissioning period. When all the testing has been completed and adjustments have been made, the machines are taken apart, packed and prepared for shipment. The end of the testing period is also known as 'Power Off'. When the machines have been shipped, installation of the machines begins at the customer's production site.

The second activity diagram in Figure 7.23 provides a more detailed description of the processes in Engineering and Manufacturing/Assembly. The Engineering process starts with mechanical engineering, through to electrical engineering, control engineering and finally commissioning. Prototyping takes place before Manufacturing. In the Manufacturing and Assembly process, mechanical installation is carried out before electrical installation, followed by control installation. A runoff check, which is a process similar to the testing and adjustment process in Lamb, is carried out and customer's approval has to be given before taking the machines apart for packaging and shipment.

Only the higher-level activities have been represented in the process model. Further decomposition was not possible (for reasons see above). For example, the "Pallet Design BP4.2.4" BP could not be decomposed into lower level activities. At the same time, there is a lack of detail in the exchanges in terms of information, resources, documents, communications, protocols between the BPs and the EAs. The information was too general to enable accurate dynamic modelling.

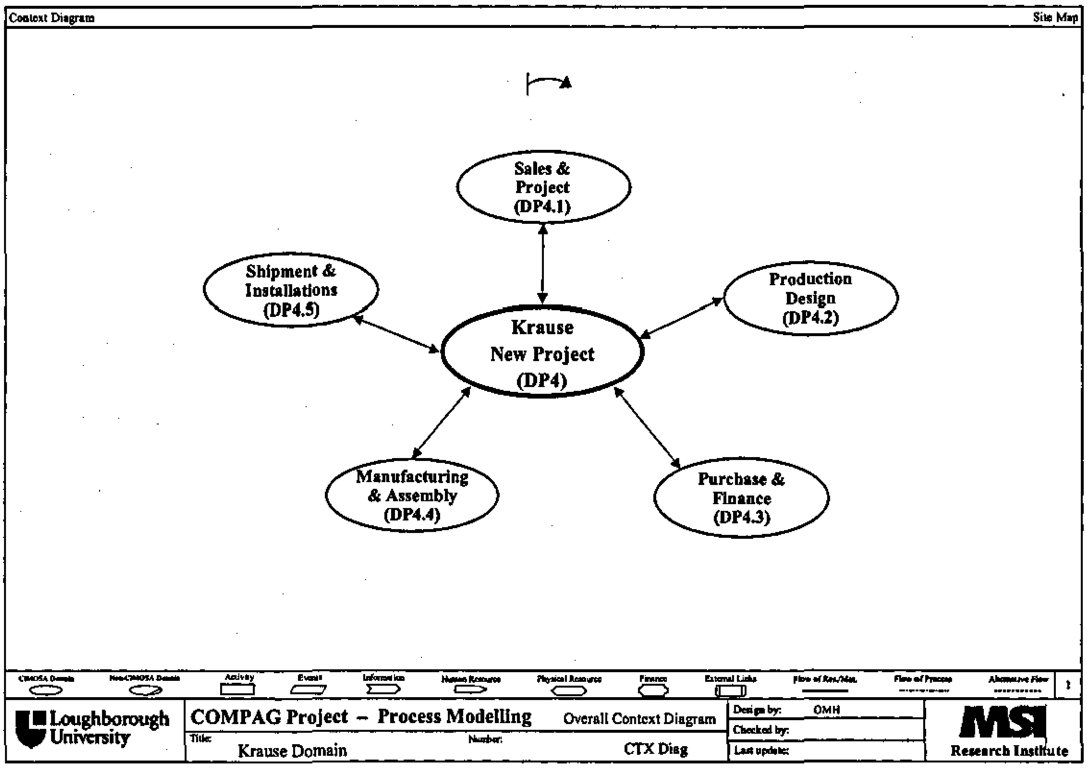


Figure 7.19 Context Diagram for Krause

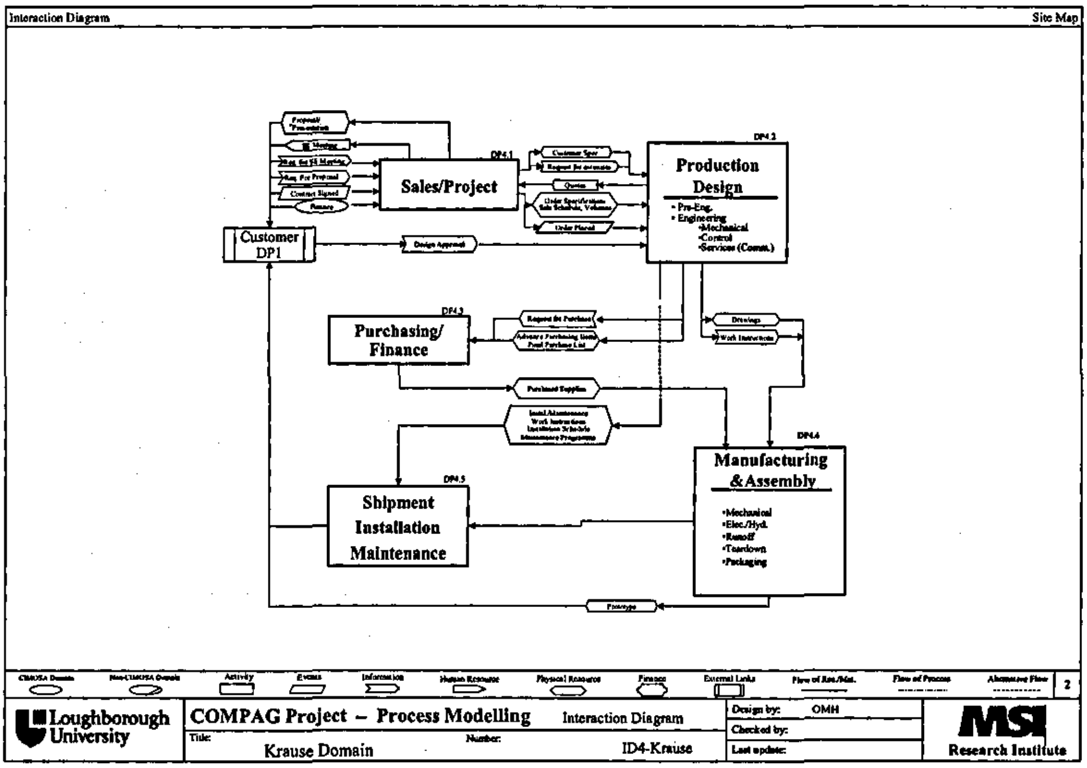


Figure 7.20 Interaction diagram for Krause

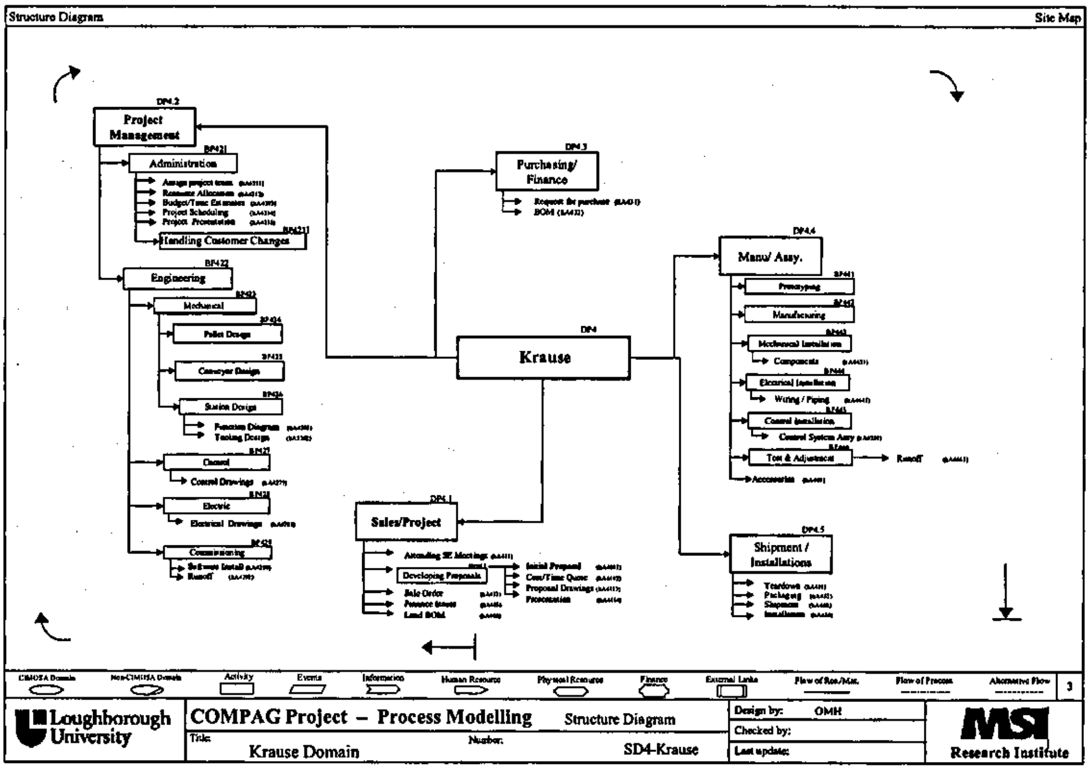


Figure 7.21 Structure diagram for Krause

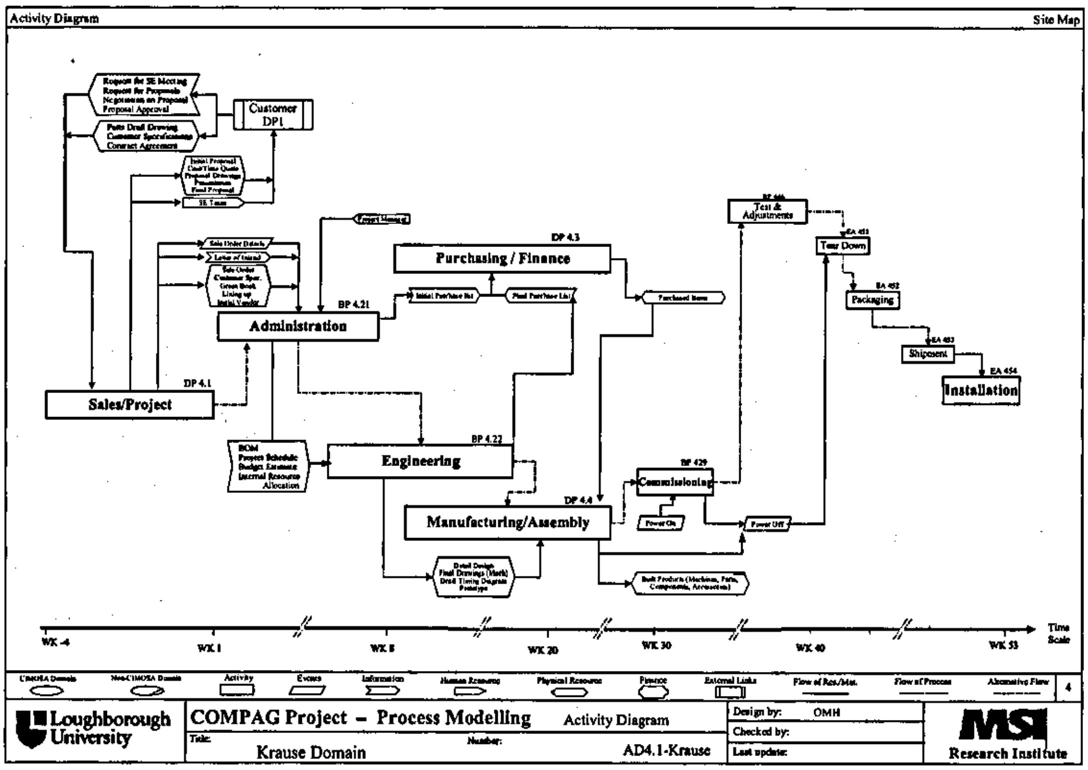


Figure 7.22Activity diagram for Krause

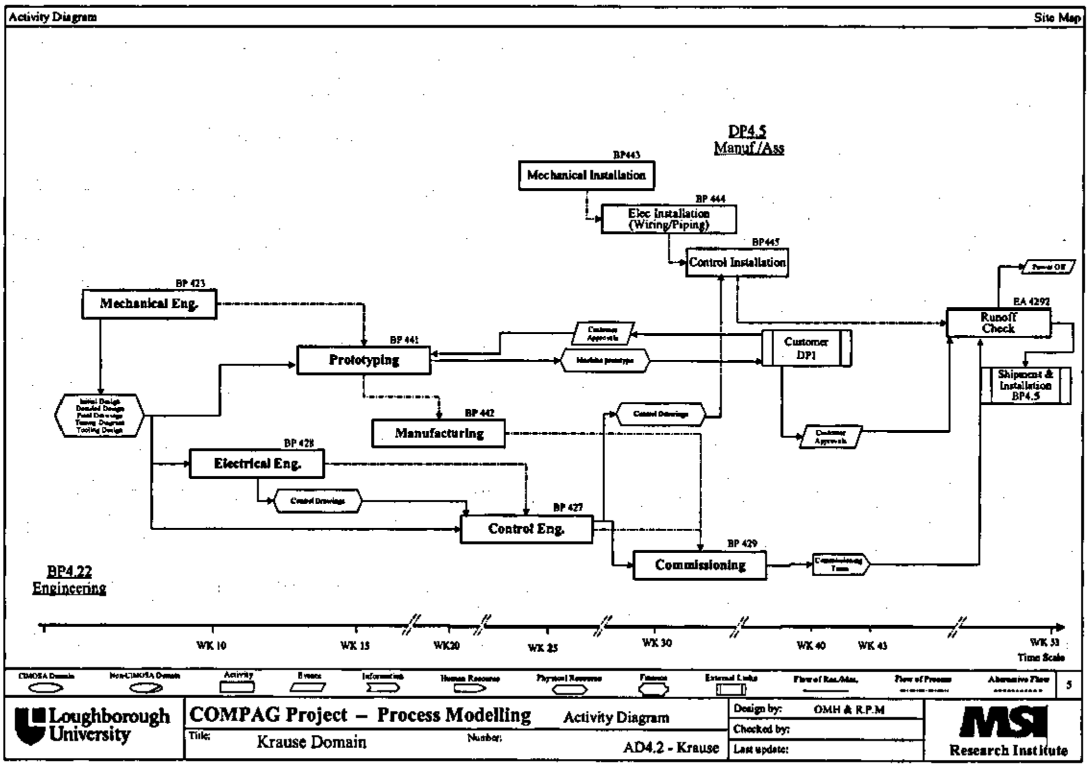


Figure 7.23 Activity diagram (2) in Krause

7.3.3 Scenario Testing

The Krause test machine is shown in Figure 7.24. It comprises a conveyor system, referred to as the transport system, and an assembly system, as shown in the 3D model in Figure 7.25. The transport system moves pallets into the assembly station where pick-and-place operations are performed on the pallet. After the operation, pallets exit the station and are sorted by a track diverter into two lanes. Radio frequency data tags (RF tags) are mounted below each pallet to carry information related to the pallet and result of the assembly operation. A total of 22 components were developed using the CBA and the control system of the test machine was implemented according to Krause's specifications.

The following sub-sections will describe the two scenario testing activities conducted using the Krause test machine.

Figure 7.24 Krause test machine

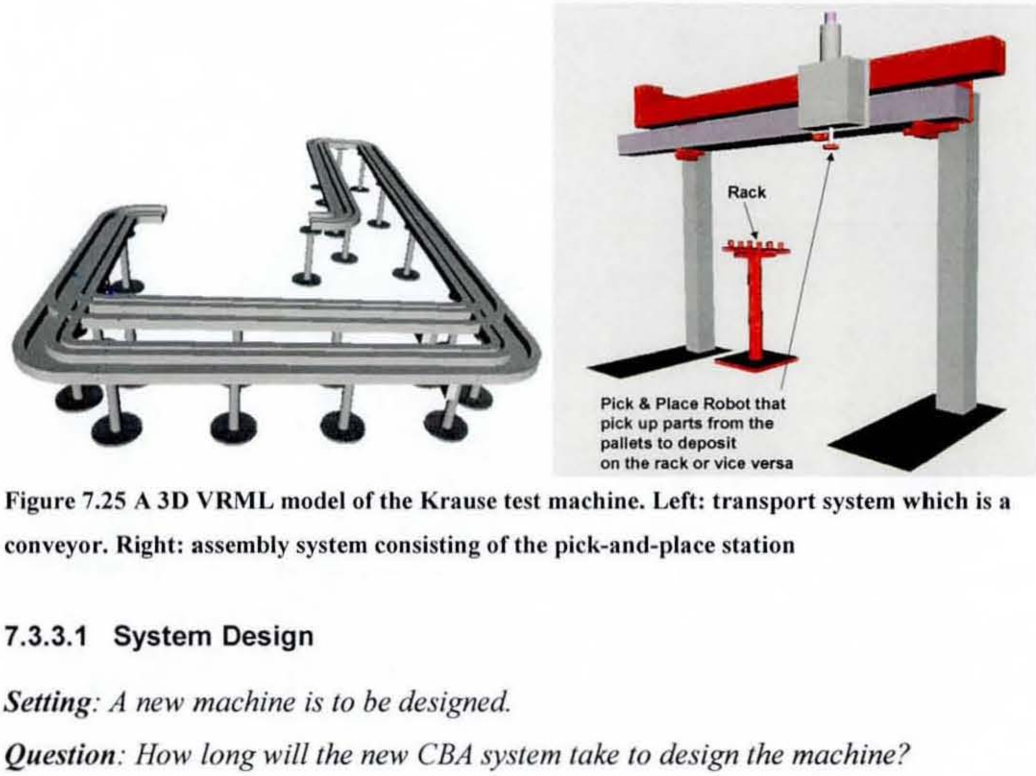


Figure 7.25 A 3D VRML model of the Krause test machine. Left: transport system which is a conveyor. Right: assembly system consisting of the pick-and-place station

7.3.3.1 System Design

Setting: *A new machine is to be designed.*

Question: *How long will the new CBA system take to design the machine?*

Metric: *Time*

In this scenario, the design and commissioning of the CBA control system for the Krause test machine was assessed in terms of the time taken for the activities to be completed. The testing was compared with the time Krause would have taken to design and commission the control system on the test machine obtained from previous project documentation and interviews with project engineers. Under the current system, designing the control logic for the transport system takes one and a half weeks, while for the assembly system, two and a half weeks are required. For commissioning, the transport system takes one and a half weeks to commission and the assembly system two and a half weeks.

The list of components implemented on the Krause test machine using the CBA is shown in Table 7.4. There are a total of 22 components which can be categorised into 11 types.

System	**Component**	***Type***
Transport System	Power Supply Unit	*1*
	Monitor Console	*2*
	Drive 1	*3*
	Drive 2	
	Drive 3	
	Drive 4	
	Stop 1	*4*
	Stop 2	
	Stop 3	
	Diverter	*5*
	RF Tag	*6*
Assembly System	Power Supply Unit	
	Monitor Console	
	Pre-stop	
	Stop	
	Fixing Unit	
	Section monitoring	*7*
	RF Tag	
	Y-axis	*8*
	Z-axis	*9*
	Gripper	*10*
	Ultrasonic sensor	*11*

Table 7.4 System decomposition for Krause test machine (adapted from Lee 2004 [71])

The time required to implement these 11 types of components are as followed:

Total component development time = Duration of 1st component + (11-1) (Duration of subsequent components)

= 11days + 10(6.5 days)

= 11 days + 65 days

= 76 days (15.2 weeks)

Assuming that there are five working days per week, implementing the 11 components from scratch will take 76 days or 15.2 weeks. If components developed based on the CBA are available commercially off the shelf (COTS), it will take two days to design the control logic for the transport system and two days for the assembly system. Commissioning of the transport system using the CBA will take five days and ten days for commissioning the assembly system. This is based on the assumption that the components were available as COTS. The HMI design and implemented have not been included. The figures are summarised in Table 7.5 and represented graphically in Figure 7.26.

	Krause	**CBA**	**CBA (COTS)**	**% Difference between Krause and CBA (COTS)**
Software Design	**4 weeks (20 days)**	**15.2 weeks (76 days)**	**0.8 weeks (4 days)**	**80 %**
Transport system	*1.5 weeks*	--	*2 days*	
Assembly system	*2.5 weeks*	--	*2 days*	
Commissioning	**4 weeks (20 days)**	**3 weeks (15 days)** [17]	**3 weeks (15 days)**	**25%**
Transport system	*1.5 weeks*	*5 days*	*5 days*	
Assembly system	*2.5 weeks*	*10days*	*10 days*	
Total	**8 weeks (40 days)**	**18.2 weeks (91 days)**	**3.8 (19 days)**	**52.5%**

Table 7.5 Summary of the time taken for designing and commissioning the test machine at Krause, using the current practice and the CBA

[17] The time required for commissioning will be the same as in the case of COTS because individual behaviour of the components had already been tested during component development. The commissioning time in this case is the time required to commission the system.

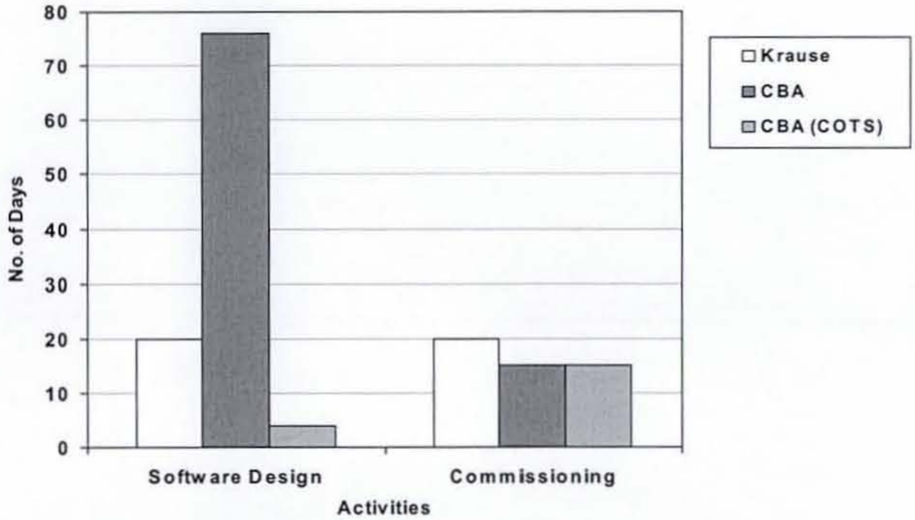


Figure 7.26 Graphical representation of the result from scenario testing on system design

7.3.3.2 System Manipulation

***Setting**: The system is processing a part and the part is removed unexpectedly. The user must now manually manipulate the system to bring it back to normal operation.*

***Question:** What needs to be done to bring the system back to normal operation in the new CBA system?*

***Metric**: Complexity*

This test was carried out from the perspective of the operator who has the responsibility to keep the machine in working order. On the Krause test machine, when a part carried on the pallet is removed, the assembly system halts because parts could not be detected. An error message appears on the HMI. In normal operation, the operator would have to change the mode of the machine from automotive to manual so as to be able to manipulate actuators to recover from the error. The operator has to read the error message on the HMI and attempt to rectify it. If this error could be rectified, the operator is able to bring the machine back to a safe "initial" position, return the machine to automatic mode and start the machine operation. If, however, the operator cannot rectify the error, he would have to call for the maintenance engineer. The above procedures are summarised in Figure 7.27.

Similar to Lamb, the recovery procedures under the CBA is identical to the above procedures since this scenario represents a standard operation procedure and must be supported by any implemented control system.

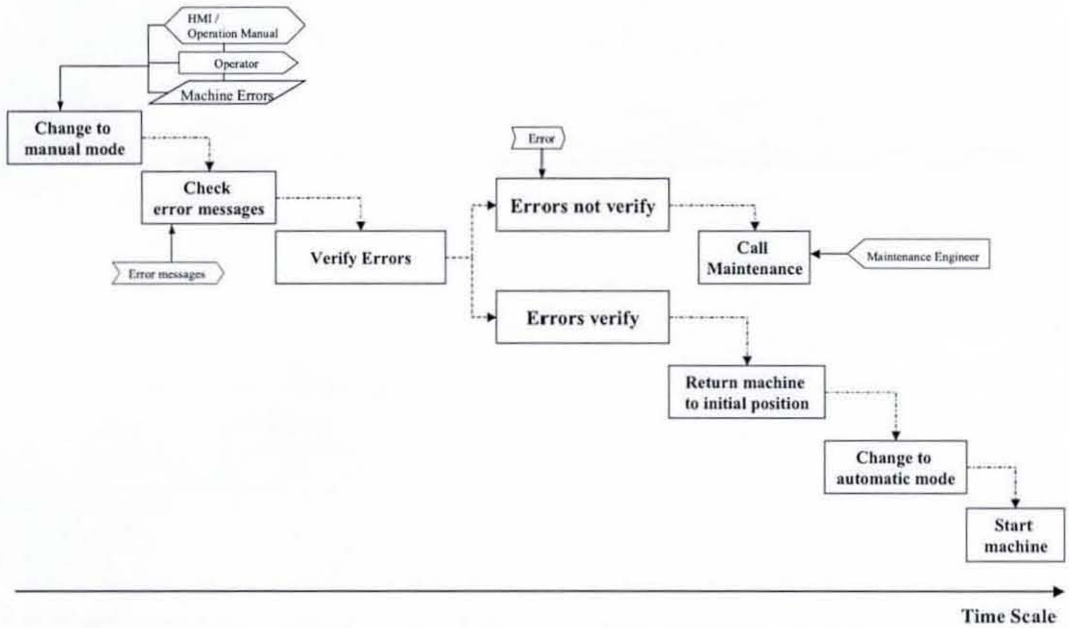


Figure 7.27 An activity diagram describing the steps taken to return the machine to normal cycle after a part has been removed from the machine unexpectedly.

7.4 Scenario Testing using Ford Test Rig

The Ford Test Rig (see Figure 7.28) is a university based test machine which is used by Ford for training its engineers on the basics of sequence logic control using PLC and ladder logic. There are five test rigs at the university that are controlled via PLC. One of the rigs has been modified for the CCG projects such that the PLC has been replaced by distributed components for proof of concept demonstrations. Some of the scenario testing which was unable to be carried out at both Lamb Technicon and Krause sites have been carried out using the Ford Test Rig because i) it is a test bed that includes more complexity in its operation than standard automotive production machinery and ii) since it is university based it is readily available for testing.

The Ford test rig consists of a feeder actuator, transfer mechanism, lift mechanism, lift pusher, conveyor mechanism, indexing table, inspection probe, drill and component picker as shown in Figure 7.29. The feeder actuator pushes a part forward to the transfer mechanism and the transfer mechanism picks up the part and deposits it onto the lift mechanism. The part is lifted up to the conveyor and the lift pusher pushes the part onto the conveyor mechanism. The part is then transported to the indexing table. The indexing table carry the part to the drilling station where a drilling operation is performed. The part is then transferred to the inspection station. When these operations have been completed, the part is picked up by the component picker and deposited into an allocated storage bin.

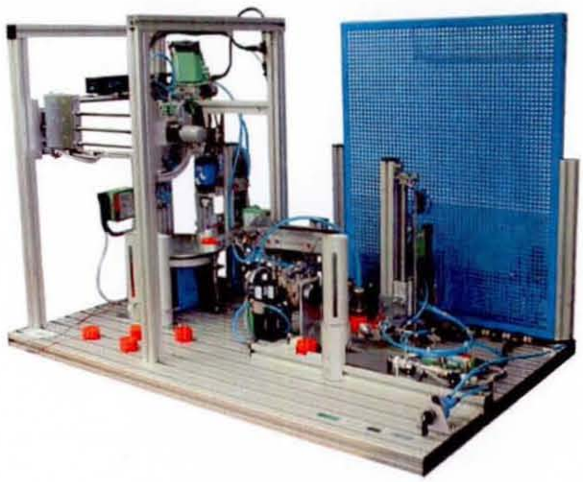


Figure 7.28 Ford Test Rig

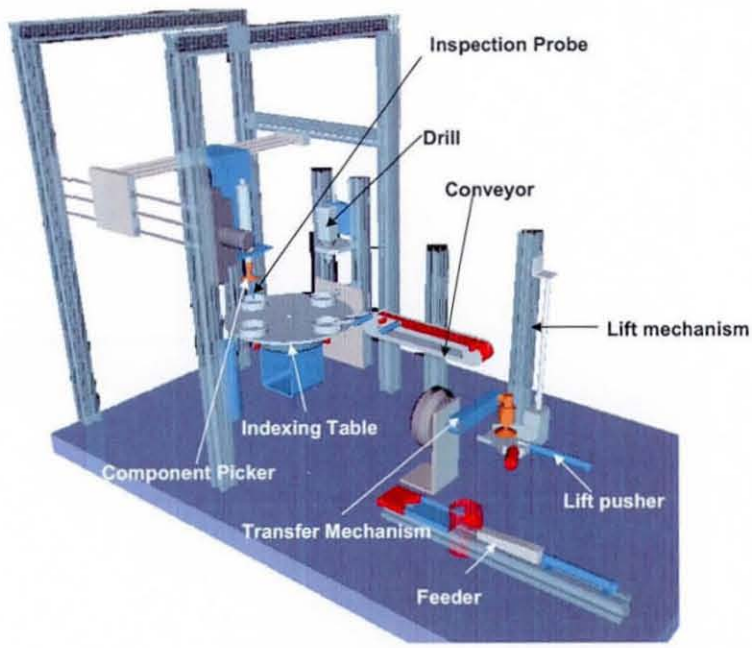


Figure 7.29 VRML model of the Ford test rig showing the different mechanism on the rig

7.4.1 Remote Diagnostic/Maintenance

Setting: *An error has occurred at the end-user's machine and the maintenance engineer needs help from the machine builder.*

Question: *How does the CBA implementation support remote diagnostics/maintenance of the machine?*

Metric: *Time*

A description of the current practice undertaken by the end user and the machine builder for machine diagnostics has been given in the Appendix E. Based on a detailed understanding of the current best practice, two user representatives were involved in the university based scenario testing, one representing a control engineer from the machine builder and the other one the maintenance engineer at the end user. The task of the machine builder's control engineer (Engineer) was to assist the maintenance engineer (User) to bring the machine back to normal operation. The remote diagnostic tools available to the Engineer were: i) telephone, ii) telephone and network video camera, iii) telephone, network video camera and web-based HMI, the Process Definition Environment (PDE) and a 3D machine simulation model implemented using the Virtual Reality Modelling Language (VRML). Random errors were injected into the Ford test rig. The list of errors injected is shown in Table 7.6.

Problem	Type of errors	Diagnostic tools	Time Taken (minutes)
1	Air supply cut off	Phone	7:05
2	Power supply cut off	Phone	7:05
3	Pairs check	Phone + video	2:46
4	Pairs check	Phone + video	4:27
5	Power supply cut off	Phone + video	6:46
6	Air supply cut off	Remote HMI + 3D models + video + PDE	1:49
7	Parts jammed on conveyor	Remote HMI + 3D model + video + PDE	3:21
8	Power supply cut off	Remote HMI + 3D model + video	5:20
9	Pairs check	3Dmodel + PDE	3:26
10	Network unplugged	Video + 3Dmodel	4:00

Table 7.6 Tabulation of the type of errors injected onto the test rig, the diagnostic tools used and the time taken to diagnose each error.

Sub-scenario 1: Using only Telephone

In this first sub-scenario, the Engineer diagnosed the problem on the machine just by using the telephone. It was found that there was a lot of conversation between the Engineer and the User. The User had to describe in detail what had happened on the machine that included the errors on the machine, the location and the position of the parts on the machine and the messages displayed by the HMI. The conversation was very much User led. The Engineer had to spend a lot of time establishing and understanding the problem. When instructions were given to the User to recover the machine, the Engineer could not visualise the progress or the effects of his recommendations. After the first and second error diagnostics, it became much easier for the Engineer to establish the problem and suggest the possible solution for the problem. The average time taken to recover the machine to a working state for the three errors is shown in Table 7.7 and Figure 7.30 shows the graphical representation of the result.

Sub-scenario 2: Using telephone and network video camera

The Engineer had support from a networked video camera in this second sub-scenario. It was observed that there was less conversation between the Engineer and the User; the interactions were more tightly focused on the root cause of the problem. The Engineer became more pro-active and could instruct the User on what actions to take based on what he saw from the video images. The time taken to diagnose the errors was shorter (-34%) than in the first sub-scenarios (see Table 7.7 and Figure 7.30).

Sub-scenario 3: Using telephone, network camera, remote HMI, PDE and 3D VRML machine model.

In the third sub-scenario, addition support was given to the Engineer via a remote HMI, a 3D simulation model of the machine and the PDE. In this scenario, the Engineer was able to determine what happened on the test rig solely from the information presented by the remote HMI and the video camera. With the HMI, the Engineer was able to give direct, clear and specific instructions to the User, and could check that these instructions were carried out correctly. The 3D machine model facilitated the verification of the parts' positions on the machine. In this test, the time taken to solve the problem reduced significantly (-49%) (see Table 7.7 and Figure 7.30).

	Average time to solve problem	Saving in terms of time (%) compared to using phone
Phone	7min05sec (425 sec)	--
Phone + Video	4min39sec (279sec)	34%
Phone + multimedia tools	3min35sec (215sec)	49%

Table 7.7 Average time taken to solve the problem using different diagnostics tools

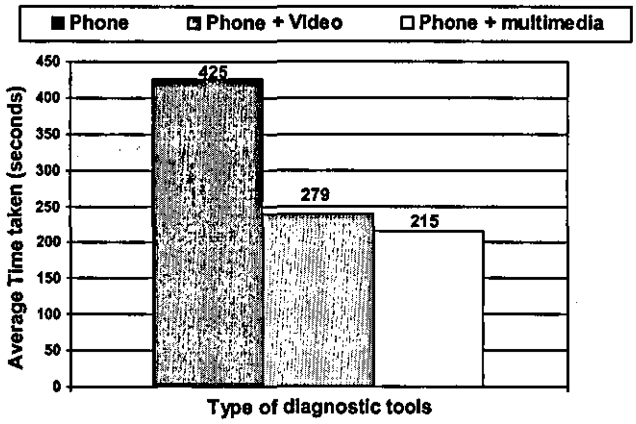


Figure 7.30 Graphical comparison of the average time taken for machine diagnostics using different tools.

7.4.2 Modification of design

Setting: *Assuming that a change needs to be implemented on the machine, what is the impact on the mechanical, control and electrical design and implementation processes?*
Question: *How does the implementation of CBA facilitate the changes in design?*
Metric: *Complexity*

The engineering tools developed under the CCG projects were not targeted to facilitate changes in mechanical and electrical design. However, the PDE coupled with the VRML and HMI environments have a range of functionalities to support control design activities.

Two tests were conducted, one in which a change (for example the transfer mechanism to remain at lift mechanism) to the control design was made to one of the original Ford test rigs controlled by a PLC, and the other one on the modified CBA test rig. Figure 7.31 shows a CIMOSA activity diagram that describes the process of modifying the control design using a PLC (i.e. equivalent to current practice adopted by the machine builders). The first step is to identify the changes required and design a solution. For PLC based systems modifications are implemented using ladder logic and a sequential function chart has to be created before writing or modifying the code. The code is then downloaded onto the test rig. If an error is found, the designer has to check, modify if necessary the code, download the code again and run the test rig to see if the changes made are correct. If errors still exist, the designer has to check the sequential charts to see if there is an error in the chart and check/modify the code. To check if the modifications are correct, the code has to be downloaded onto the test rig and operated to determine if the sequence is correct. If there are no errors in the code in the first instance, seven steps (i.e. identify changes, design solution, modify sequential charts, modify code, download code, run test rig and no errors found) are required to complete the commissioning. If however an error exists, there are two possible sources: either within the sequential chart or in the code. If it is just one error in the code, 13 steps are required before completing commissioning (i.e. identify changes, design solution, modify sequential charts, modify code, download code, run test rig, errors found, check software code, errors found, check/modify code, download code, run test rig and no errors found); if the error is within the sequential chart, 14 steps are required (i.e. identify changes, design solution, modify sequential chars, modify code, download code, run test rig, errors found, check software code, no errors found, check/modify sequential chart, check/modify code, download code, run test rig and no errors found).

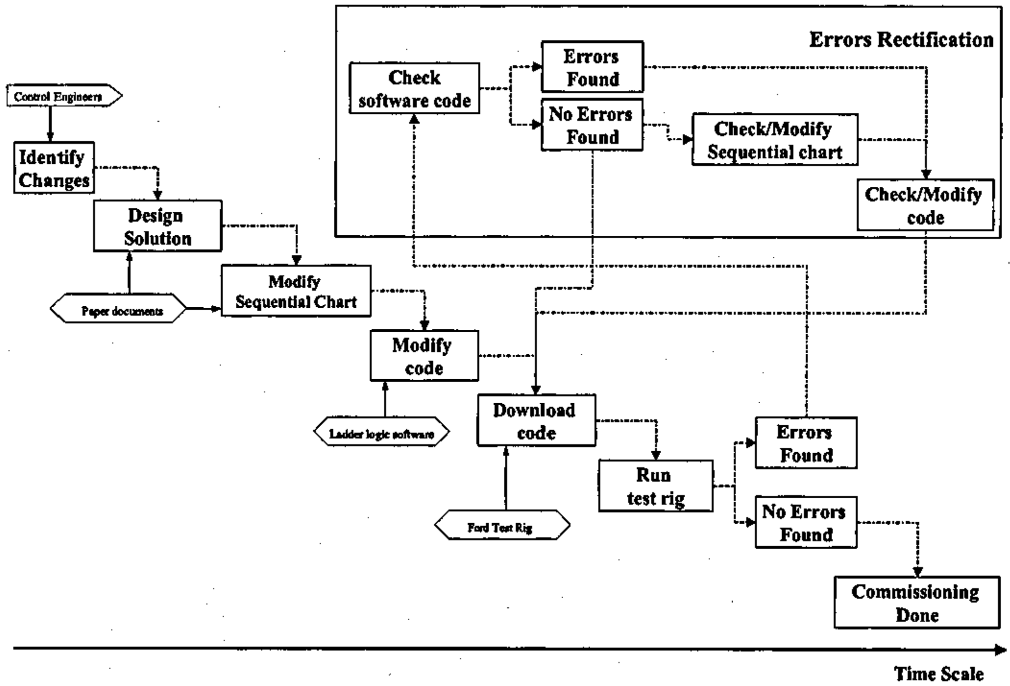


Figure 7.31 An activity diagram describing the processes of modifying the control design under current practice.

The activity diagram in Figure 7.32 describes the process of modifying the control design under the CBA adopted by the CCG projects. As in previous example, the first step is to identify the changes required and design the solution. In the CBA case, the state machines are located within the components, sequence logic is contained in the interlocks. The design of the solution is followed by running a simulation of the control logic created. If no error exists, the interlocks can be downloaded onto the test rig. If an error exists, the designer has to check the interlocks, modify and run the simulation again before downloading the interlocks to the test rig. It took eight steps to complete commissioning; and if for example errors exist, ten steps are required. In addition, the CBA system provides visualisation tools (i.e. 3D VRML machine representation) that can be used to aid the control engineer in diagnosing errors. In the PLC based control system development, the control engineers have to rely on their experience to determine how the control code will affect the actual operation of the machine. By using the editor (i.e. PDE), pre-download simulation can be carried out and pre-knowledge of component behaviour is made possible.

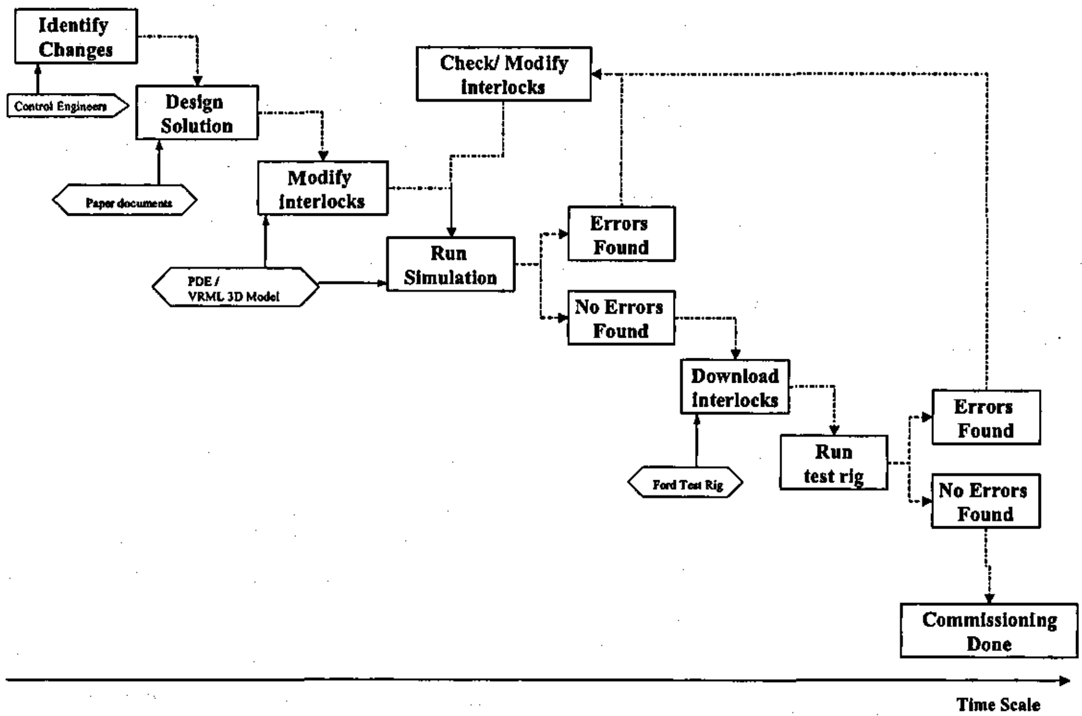


Figure 7.32 An activity diagram describing the modification of design under CBA

7.5 Survey Result

The questionnaire sent out to the project collaborators has been discussed in Section 5.5. The aim was to determine quantitative and qualitative opinions on the CBA from technical and management personnel within the industrial collaborators. However, the response rate for the survey conducted was poor. This was not totally unexpected since only a limited number of engineers at the collaborator sites were aware of the CBA activities and all of these were at a senior level of technical and management capability. Of the ten targeted respondents, only three responded. Due to the low response rate, the results from the survey could not be generalised and could only be regarded as the individual opinion of the respondents. It was not appropriate to undertake a detailed statistical analysis; the results were complied manually.

Of the three respondents, one of them is a component supplier, one machine builder and one end user. As mentioned in Chapter 5, the questionnaire contained five sections on the different features of the tools development under the CCG project: a common model environment, control environment (embedded control), engineering environment (Process Definition Environment PDE), visualisation environment (3D VRML modelling) and a human machine interface (HMI) environment. A summarised analysis of the results is given in this Chapter. The complete questionnaire responses are given in Appendix B.

On the whole, the stakeholders were able to appreciate and acknowledge the advantages of distributed control. However, the main driver behind any adoption of the technology would be that distributed control technology should lead to an overall decrease in cost and time. The machine builders and end user do not expect this new technology to increase the skills requirement for control and electrical engineers. However, for the component supplier, the job responsibilities of both control and electrical engineers are expected to increase since the components have to be developed with increased control and hardware and software interface capabilities.

For the design and engineering tool (i.e. PDE), only the end users and machine builders are interested in getting such a tool since it is an important tool to them, while the component supplier showed no interest in it.

All of the three stakeholders believe that the 3D virtual representation and animation of the machine is important to mechanical and control design. They also believe that 3D models would be useful for SE and remote diagnostics. As for the impact of having such a tool, the three stakeholders were divided in opinion; the machine builder believed it would have no effect on cost, time, skill and human resource requirement; the end user believed it will lead to a reduction in cost and time; the component supplier felt that it will reduce the cost but has no effect on time. The main concern on having such a visualisation environment is cost and time required to develop the model and the retraining required to ensure that the tools could be used effectively.

As for the Human Machine Interface (HMI) environment, the respondents expected that the web-based HMI should not affect the skill and human resource requirement. In terms of the cost and time required for machine building, opinions differed. The machine builder believed that it would have no effect; the end user believed that it would reduce the cost but would not affect the time required; the component supplier held the opposite opinion to the machine builder, i.e. time will be reduced but cost will not change. The machine builder is more interested in having the Web-based HMI than the end user and the component supplier since it is the machine builder's responsibility for initial HMI development and remote diagnostics for error recovery and maintenance which would both be aided by Web based HMI development.

7.6 Summary

Three case studies for the evaluation approach developed in this research have been discussed in this Chapter. The data collection methods, scenario testing and enterprise modelling approaches have been implemented in the first two industrial case studies, while the third case study, i.e. the university based test rig was used mainly for scenario testing. A summary of the results from a questionnaire survey has also been presented.

In the two industrial case studies, descriptions of how data were collected and represented using the enterprise modelling approach have been given in each case. A set of CIMOSA diagrams has been created for each of the case studies which allows the comparison of the AS-IS and TO-BE approached to system development and maintenance. Dynamic modelling was considered in one of the case studies. The data collected were input to scenario testing. Not all the scenarios deemed relevant to the evaluation of the CBA could be tested in the two industrial case studies, which then called for the need to introduce the third case study to test the remaining scenarios. The third case study is more of a laboratory setting and the test machine used is a training set used by the end user for training their engineers. In the survey, response rate was very low. However, there was feedback from at least one of each of the stakeholders (i.e. end user, machine builder and component supplier). This feedback was summarised and analysis was given. Detailed discussion and analysis of the case studies and the results can be found in the next two chapters.

Chapter 8

Research Result and Analysis

8.1 Introduction

In this Chapter, the results from the case studies are discussed. Quantitative and qualitative analysis is presented. Based on the analysis, observations are made on the perceived benefits of CBA and the issues associated with its implementation. These observations are discussed from four perspectives: product, process, business and people.

8.2 Research Analysis

The results of the evaluation exercises (i.e. interviews, analysis of documents, scenario testing, survey and case studies) can be broadly classified as either quantitative or qualitative.

8.2.1 Quantitative Analysis

Based on the dynamic modelling of the AS-IS and TO-BE process models, the implementation of the CBA for the design and build of engine production machinery has the ability to reduce dramatically the project development time and hence costs (section 7.2.4). For example the development time could be readily reduced from 52 weeks to about 49 weeks, i.e. a savings of three weeks. A 52 weeks project would cost approximately £20 million[18], i.e. about £380,000 per week, a saving of three weeks would result in a saving of £1.14 million. It would also imply that potentially, the end user would be able to bring forward the start of "Job 1" (see section 2.2.2) and thus introduce the product earlier onto the market. This saving in time is primarily enabled by the embedding of the control and electrical systems within the components, and hence reducing the control and electrical engineers' time in designing these aspects of the system.

In addition to the design and build advantages, the implementation of the CBA offers the opportunity for remote diagnostics. Although currently, machine builders already

[18] This figure is based on interviews with Lamb Technicon.

have the capability to download and upload PLC code from a machine remotely, they lack the appropriate tools to enable a complete visualisation of the machine state in terms of the machine logic, video footage and VRML representation of the physical components states prior to the occurrence of the any errors. From the evaluation conducted in this research, it has been found that having visualisation tools to aid in remote diagnostics would reduce the time taken to diagnose typical machine errors (by about 50%, see Section 7.4.1). This can be explained by the fact that a lot of time is currently spent on establishing the problem on the machine. At the moment, without the support of visualisation tools, diagnosis is only effective when the machine builders engineers are onsite, which is costly in terms of time, expenses and detrimental to other development projects that the engineers could be working on. In addition, the quality of the information given to the machine builders depend on whether the end user machine operators and / or maintenance engineers can provide enough detail and give a good description of the problem, increasingly rare considering the move towards lower salaries (and hence lower educated personnel) and de-skilling (i.e. multitasking) within certain sections of the industry.

It has also been observed that CBA can also help to reduce the complexity involved in changing the control design. As shown in section 7.4.2, for PLC design, a sequential chart has to be designed before designing the logic code. If any errors exist in the control logic, it could be in the sequential chart or within the code. With CBA, the embedding of control behaviour within the component facilitates the creating of a simulation tool for control logic (see Section 4.2.3) which enables the logic to be verified prior to downloading to the actual machine.

8.2.2 Qualitative Analysis

Qualitative analysis has been made with reference to the findings from the scenario testing and the survey results. The research has illustrated that implementing CBA can help to reduce the process of document translation that is currently undertaken (section 7.4.2). As discussed, the translation process is vulnerable to human errors and takes up unnecessary resources. An elimination of the translation process will help to improve work efficiency, reduce errors, and reduce cost and time.

The design of the control system under the CBA also benefits from the integrated engineering tool that has been developed to support this approach. Scenario testing (section 7.4.2) has shown the complication involved in error checking of the control design under current practice in which the control logic has to be downloaded onto the machine to check if the logic is correct. Under the CBA, simulation tools for verifying the control design have been made available to the control engineers. This enables the reduction or even elimination of errors in design, and helps to reduce the burden during commissioning of the machine. This view is shared by all of the three stakeholders in the design and builds process. In the survey, all of them indicated that they felt that having engineering tools such as those developed for the CCG projects (i.e. PDE, HMI and 3D visualisation), would reduce the time and cost of commissioning. The end user and the machine builder, who would be the main users of the engineering tools, are both keen to utilise these tools within their businesses.

The stakeholders are divided on the issue of change in skills requirements for control system design under the CBA. The end user felt that the CBA approach would increase the skills requirement for control design, the machine builders believed skills requirements would decrease and the component supplier felt that there would be no effect. This could possibly be explained by the fact that since CBA is a new technology, the end user felt that its engineers would have to pick up new skills; from the machine builder perspective, the embedded control technology would imply that there is less design work for the control engineers and thus less demanding work for the control engineers. This, however, does not imply the job responsibilities of the control engineers will be lower, in fact, they are expected to remain the same. The component supplier acknowledges that job responsibilities of control engineers would be higher because of the need to embed the control logic within the components, although the skill requirements are not expected to change.

From the point of view of machine implementation, the CBA is able to meet the machine specifications and requirements of the stakeholders (Section 7.3.3.1 and 7.3.3.2). The two test machines have been implemented according to the specifications of the machine builders (i.e. Lamb Technicon and Johann A. Krause) and have passed detailed commissioning tests set by each of the two machine builders. The scenario testing has demonstrated that the steps taken to put the machine back to normal cycle are the same under both current practice and CBA. Furthermore it is unlikely that the implementation of the CBA will affect the operation of the machine since the input requirements for the machine operation are obviously the same in all cases. However, more detailed testing is required to assess system manipulation from the point of view of other users, such as maintenance engineers, control engineers and mechanical engineer to ensure that this is the situation in all cases.

The research had also indicated that there is limited support for mechanical and electrical design and commissioning activities. At the moment, the purpose of the 3D models created in the CCG project is for simulating the machine control processes, enabling diagnosis of errors and supporting training or maintenance, but not to support mechanical design activities (such as performing mathematical calculation of forces, stress, pressure etc). The 3D VRML machine representation has the potential to support mechanical design activities, but in order to do that, the model has to be created at a higher level of detail and precision. With the current capability of the 3D visualisation environment provided under CCG project, while the end user felt that it will help to reduce the cost and time of machine building, the machine builder believes it will have no effect on the cost and time of machine building. This is because the machine builder felt that the 3D visualisation tools have to be integrated with their in-house tools so that these visualisation tools can support the control design activities.

Based on the results of the survey, the end user and machine builder believe that the embedded control technology (such as CBA) would help to reduce cost and time. From the component supplier's point of view, however, embedded control technology will reduce time but not cost. This could be explained by the fact that flexibility and reusability are being associated strongly with the technology but the cost of producing the item might not be necessarily lower. In terms of the human resources required, the end user sees the need to increase human resources, but the machine builder and the component supplier do not share the end user's view. The CBA is not expected to change the job responsibilities of the engineers, except for the control and electrical engineers from the component suppliers. Their job responsibilities are likely to increase. In terms of the different engineering tools that have been developed to support CBA (i.e. PDE, visualisation tool, HMI generating tool), the end user holds the opinion that they will help to reduce the time and cost. The machine builder and the component suppliers do not see all tools helping to reduce time and cost; some tools will help to reduce time but not cost and vice versa. In general, the end user and the machine builder are keener to have the tools, but the component supplier is not so interested in them. This is understandable because the tools are more applicable to the machine builders and the end users than to the component suppliers.

8.3 Research Observations

Based on the evaluation studies that have been conducted, benefits and issues associated with the implementation of CBA in the automotive sector have been identified. Lessons have been drawn from the component-based software engineering (CBSE) development, as it is the closest example to the use of the CBA. Some of the benefits and issues raised in the software industry are of common concerns to the automotive sector.

The potential benefits of implementing CBA have been outlined in Chapter 4. However, in a highly complex project such as the production machine development project, there are a number of companies collaborating with each other to produce the final product. Each of these collaborates i.e. stakeholders may have different objectives for the system. It is important that in discussing the issues associated with CBA, these issues are discussed not only from the perspectives of the individual stakeholders, specifying the advantages and issues associated to each stakeholder, but also from the different perspective of system development.

Figure 8.1 is a summary of the issues and benefits associated with implementing CBA for the stakeholders. To be consistent with the component-based evaluation undertaken in the software engineering industry the author has adopted four viewpoints to discuss the different perspective of system development, which are: product, process, business and people (see section 2.3.5). Product views are concerned with the product i.e. the components and the production machines; from the component supplier's point of view, that is the components which are manufactured or supplied; from the machine builder's point of view, i.e. the production machine that has been produced and from the end user's point of view, i.e. the machine that has been specified. Process views are concerned with the working process; business view looks into the business issues and people issues are related to the human associated issues.

8.3.1 Product Issues

From the product perspective the end user and machine builders will benefit from product assurance, increased agility and reduction in product cost and time. In addition end users will benefit from increased reusability. The component suppliers will benefit from being able to offer products with better functionality but they will have to develop the hardware and software components. There are several issues which are of concern to the stakeholders, namely, interoperability, support for reuse, component library, political issues and development of 3D models.

Component Suppliers

Develop Hardware and Software Component

The component suppliers will have to provide significant changes to their products. Not only will they have to supply the hardware, they will also have to supply the software of the components. They will have to pre-programme the basic operations of the components and encapsulate the functions. Thus, the job responsibilities of the component suppliers' control and the electrical engineers are likely to increase (see Appendix B).

Products with Better Functionality

Although the component suppliers will see an increase in their responsibilities, however, they will be able to add value to their products by embedding increased functionality into the components. Component technology increases the level of abstraction and more complexity could be packaged into a component framework. Better component diagnostics features such as component's mean time failure, performance data, error history could be embedded into the components which would allow the user to retrieve these information readily. This information would be useful for the maintenance of a machine.

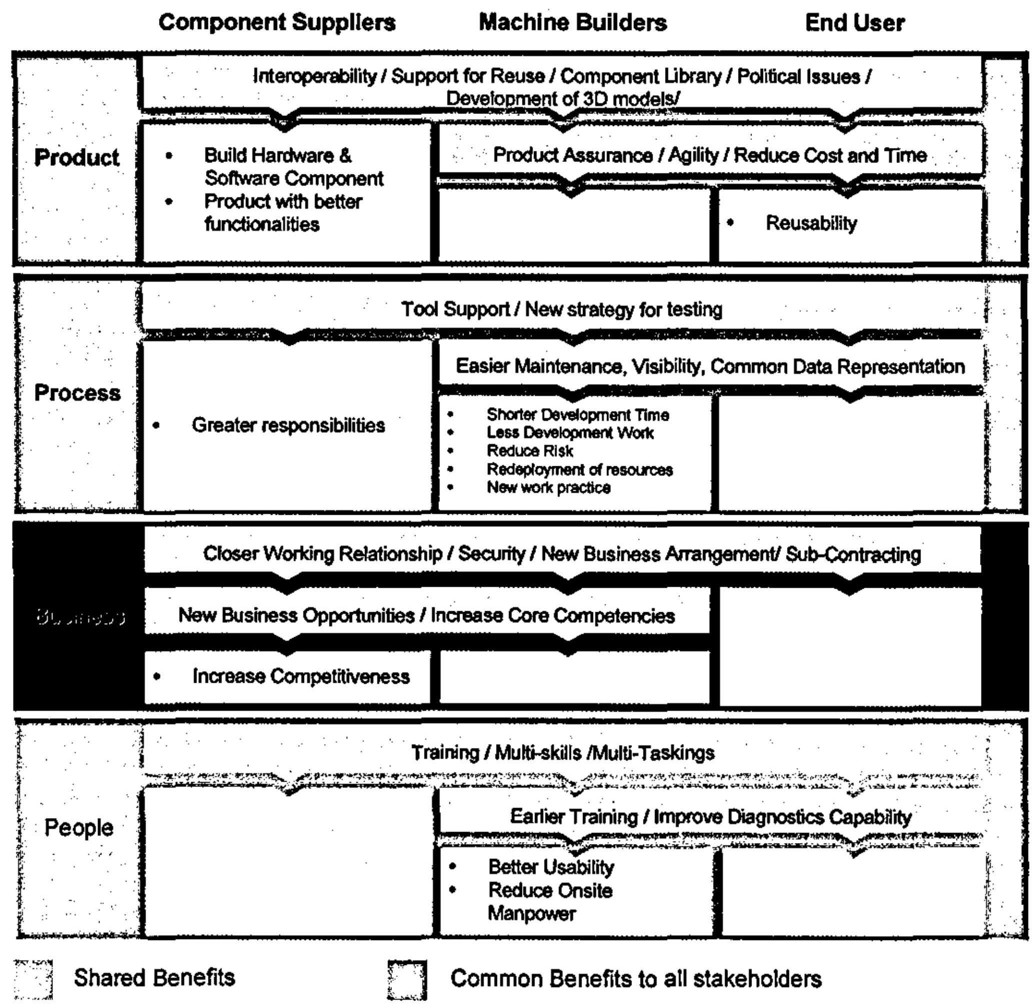


Figure 8.1. Perceived benefits from the implementation of CBA in the automotive sector with reference to the stakeholders (adapted from Brereton and Budgen 2000 [41]).

End Users

Reusability

As discussed in Chapter 2, competition has forced the end users (i.e. car manufacturers) to shorten the time taken to launch new products. The end users would like to be able to reconfigure or reuse their machine such that the machine can be used to produce different models of car. Theoretically, the availability of the component technology would enable the machine builders to reconfigure the existing machine to fit the requirements for new models of automotive engines.

Common Issues

Interoperability

Currently, the components from different suppliers are not interoperable. Traditionally, the component suppliers have been supplying proprietary solutions to maintain and increase their market share. However, it has been the machine builders and the end users hope that the components would be interoperable, which would enable them to have flexibility in their choice of components. Therefore, there has to be an agreement or common understanding in the industry to encourage interoperability of products. In other words, there is a need to establish an industry common standard for components.

Support for Reuse

One of the key attractions of the components is reuse. Although the machine builders are concerned about the life-span of components in relation to reuse, it is the desire of the end user to be able to reuse as much of their existing machinery as possible. Currently, the machine builders are already providing services for modification and extension of existing systems. The components suppliers need to identify components which are suitable for reuse, or to provide certain support to the machine builder for the reuse of suitable components. Theoretically, the existing machine could be reconfigured under the CBA to fit the production requirement for new models of car, however, the life span of the physical components has to be taken into consideration. Such reuse of components might be restricted to certain hardware that does not wear out easily. There have to be incentives for all the three parties in order to encourage reuse (see section 2.2.3).

Component Library

There are two possible ways of developing a component library for the machine builder, either they build their own library or the component suppliers supply it. One of the machine builders had suggested that the end user should specify the standard for component libraries the component suppliers to comply with.

Political Issues

Feedback from the stakeholders indicates that, in general, all parties can appreciate the benefits of the CBA. The machine builders, in particular, are in favour of CBA development because it will greatly reduce their workload and development time. They believe that if the end user is in favour of this development, it would give the component suppliers the incentive to meet the demand. The end user, however, is very cautious in adopting new technology because any such investment is for ten or more years. The end user wants assurance from the machine builder and the component suppliers about the viability of the technology. Component suppliers have their own concerns; they have their proprietary solutions and any new technology has to be built upon their current technology in order to protect their market share. They insist that if the end user is willing to commit to the new technology, they would proceed in the development of component libraries (see above). As a result, deadlock has occurred in which no stakeholder is willing to take the first step unless they have assurance from the others. Thus, to realise the CBA in the automotive sector, it is important that the three stakeholders can come to an agreement on these issues.

Development of 3D machine models

From the survey results (section 7.5), the end user is concerned with the resources and the time required to develop the 3D models. The machine builders have indicated that they are not too keen on investing the extra resources to develop 3D machine models (see Survey results Appendix B). The component suppliers have not indicated whether they are willing to provide the models for the machine builders since they, too, would need to invest extra resources into producing the models for the components. Having said this, however, the parent company of one of the machine builders had already started using 3D models for machine design. These models had been used to identify possible problems and simulate possible solutions [177]. Therefore, individual company will have to see the benefits to invest into the technology. For the sector as a whole, if there is a strong demand for the use of this technology, there will be supply to meet the demand.

Machine builders and End user

Product Assurance

Components that are reused many times have higher quality and reliability than newly coded components. Through the reuse of the components, the stakeholders will have the assurance of the quality of the components (see section 2.3.3).

Agility

The reconfigurability of components enables the end users and the machine builders to enjoy greater degree of agility. End users would like to be able to reconfigure their machines to produce different models of engine. Currently, this is made possible by planning and designing the machine such that extra stations could be added on or tool changes could be made to meet the change requirement. By using the process simulator within PDE, the machine builder could conduct tests to assess the feasibility of implementing new control software that is required for the new machine configuration.

Reduce Cost and Time for machine design and build

From the survey results, it has been shown that the end user and the machine builder believe that embedded control technology would help to reduce cost and time. It is not just the technology alone that is associated to this saving. The engineering tools that had been developed to support the technology are expected to provide similar benefits to the stakeholders during machine design and build although at varying degree of applicability. At the same time, it is likely that shorter development time would reduce the cost of development and thus lead to an overall reduction in production cost. (see Appendix B for Survey results)

8.3.2 Process Issues

The machine builders and the end users are the main beneficiaries from the process perspective. They both share some common benefits and the machine builders in particular will benefit from a shorter development time, less development work and a lower risk in development. However, the machine builder needs to consider new work practices and redeployment of resources. The main issue that is of concern to the component suppliers is increased responsibility within the machine design and implementation process.

Component Supplier

Greater responsibility

It is foreseen that the component suppliers will play a greater role in the process of machine design and build. Currently, the component suppliers assist the machine builder in the selection of suitable components and the programming of special equipment such as servo drives. With the implementation of the CBA, their responsibilities will not only be limited to special equipment; they will also become responsible for the control behaviour of every individual component. It follows that the time reduction in the machine building process and the machine builder's workload are compensated by increases to the component supplier's workload and additional responsibilities.

Machine Builders

Shorter Development Time

With the embedding of the control and electrical functions in the components, it is envisaged that the machine builders will be able to reduce their development time in these two areas. This will lead to an overall reduction in the machine development time as shown from the result of the dynamic modelling evaluation (see section 7.2.4). With the availability of machine process simulation and visualisation tools, the machine builder will be able to simulate the machine sequences at the early stages of the project and any necessary rectification could be identified as early as possible. This will reduce the complexity of the final commissioning and the cost and time associated with commissioning is expected to reduce. On the whole, shorter development time for the machine builder will imply that the production cost could be reduced. (see Appendix B for Survey Result)

Less Development Work

The engineering environment created to support the implementation of the CBA (i.e. PDE) offers the ability to create a timing diagram, and from which, the control logic of the machine could be defined and configured. This not only eliminates the process of translating the timing diagram into control logic, but also enables the configuring of the machine logic at the same time. The control engineer would only have to design the control logic to assemble the various components in the machine, instead of having to write the programme code for each of the devices that make up the machine. There will be less development work for the machine builders.

Reduce Development Risk

As mentioned earlier, the process of machine design and build is highly experienced based and there is a lot of reuse in a "white-box" [43] manner. As one of the machine builders has indicated, about 70% of their control functions are repetitive, however, the ability to reuse these previous work depends on whether anyone in the team could remember whether a similar functions had been used in previous projects. The component technology would enable the machine builders to reuse previous work systematically through the compilation of a component library. Reusing these components which have been proven in previous projects would minimise the unknowns and reduce the development risk of the machine builders.

Redeployment of resources

Currently, commissioning of the machine is performed by one commissioner. To be able to sub-commission the sub-assemblies of machines and to shorten the development time, machine builders need to reallocate the manpower for the tasks. At the same time, if the machine builders are to provide remote diagnostics services, they will need to re-deploy their human resources to provide 24 hour support.

New Work Practices

The machine builders, as mentioned earlier, would need to develop new working practices for the assembly of mechanical parts and commissioning. They would not have to wait for the mechanical system to be fully built before they could undertake commissioning and they would need to organise the associated control and electrical design activities such that they could be on time for sub-assemblies. Ideally, the machine builders would be able to "select/design/implement", just like the component integrators in the software industry (see 2.3.5.2). They would select the required components, followed by designing and implementing the new machine.

Common Issues

Tool Support

The new approach require new tools to support the activities. Opinions were divided on who should provide the engineering tools required. Some stakeholders think that the end user should specify an engineering environment which all the component suppliers would comply to; some suggested provision by a third party so that it is independent of suppliers; others suggested it should be provided by the control supplier. Nevertheless, the machine builders would like the tool to be integrated with their existing tools, the end user would like to have the tool available commercially or alternatively, partnership with suppliers; the component suppliers think that the tool has to have full worldwide service support.

New Strategy for Testing

Both the component supplier and the machine builders need to develop new strategies for testing. Not only do they need to ensure that the mechanical parts of the product are in good working order, they have to test the software as well. The machine builders, if they are performing sub-commissioning of sub-assembly systems, will need to develop suitable strategies for the purpose. They will need to learn how to diagnose a machine and to identify problems under the CBA.

Machine builders and End users

Easier Maintenance

Maintenance is expected to be easier since with the component technology, the problems can be isolated and dealt with without affecting the rest of the system. Any malfunctioning components could be replaced easily with a new working component (i.e. plug-and-play). With the availability of web-based HMI and the 3D virtual representation of machines, machine builders would be able to implement remote diagnostics for the end users. As shown in the scenario testing, the time taken to diagnose a problem with appropriate visualisation tools can be dramatically reduced. At times when it is necessary for the machine builders to go on site, their engineers will be able to login remotely to the machine, assess the machine and then if necessary go on site with more detailed knowledge of what happened on the machine and what the possible solutions might be.

Visibility

The availability of the 3D virtual representation of the machines will improve the visibility of the development at various stages of the lifecycle of machine design and build. The machine builders have expressed their interest in using the models for marketing and sales purposes; they will be able to produce a conceptual model of the machine for presentation to the end user. Based on the model, alterations and improvement could be made progressively. The use of the same model during SE as the final machine would also enable the participating engineers to visualise the machines that are to be built. This will help to clear up any misconceptions about the machine to be built. At the building phase of the machine, the model could be used to simulate the process and identify potential errors/problems in the machine sequences. When the machine has gone into production, the model can again be used to support maintenance.

Common Data Representation

A common representation model is made possible through the use of the component technology. The same set of data (i.e. component data) once entered into the common depository, is available for use by the different users. This would ensure data consistency and reduces the chances of errors from translation and interpretation during the lifecycle of the machine (i.e. from design, build, production to maintenance)

8.3.3 Business Issues

From the business perspectives, individually, the component supplier is likely to be able to increase their competitiveness. Between the component suppliers and the machine builders, they will be able to benefit from having new business opportunities and ability to increase their core competencies. The three stakeholders will be able to have a close working relationship. The common issues that are of concern to them are, namely, security, new business arrangement and sub-contracting.

Components Suppliers

Increase Competitiveness

There is a greater opportunities for the component suppliers to increase their market competitiveness by adopting the component-based approach. They would be able to distinguish themselves from their competitors by improving the functionalities of their range of products. Various functions and diagnostic capabilities can be embedded into the components, such as for example mean time between failure of components (MTBF), performance data, error history and maintenance history.

Common Issues

Closer Working Relationship

A closer working relationship will be formed between the three stakeholders. This will enable them to share information, new technologies and products, critical resources, new markets and core competencies, thus enhancing the competitiveness and improving the responsiveness of each other.

Security

Firstly, with the use of common data representation, i.e. sharing information through the common representation, the stakeholders are concerned about outside intrusion into their corporation network. Secondly, the potential threat of unauthorised access to their machines arising from remote diagnostics capabilities is a major concern to the end users. Such concerns require the stakeholders to address the issues and work out solutions that are agreeable to all parties. Possible solutions suggested from some of the stakeholders are: i) having separate network for machinery, and ii) restricted access users.

New business arrangement

With the new business opportunities available to the component suppliers and the machine builders, there need to be new business arrangements. For example, in providing support for reuse, the component suppliers and the machine builders need to have a new agreement on how the component suppliers would support this service. The machine builder would also be able to negotiate a new service contract with the end user if they are to provide remote diagnostics support for the end user.

Sub-Contracting

Most of the machine builders sub-contract part of the work at various stages of the lifecycle of machine design and build. If the CBA is to be implemented, these sub-contractors will be affected as well. For example, if the end users require that the machine design is to be in 3D format, the sub-contractor who does the mechanical design for the machine builder will need to have the same capability. Moreover, the engineering tools that the sub-contractors are using may not support or be compatible with the CBA. In this case, some translation of documents will be inevitable.

Component Suppliers and Machine Builders

New Business Opportunities

New business opportunities are available to both the component suppliers and the machine builders. The component suppliers are able to offer new products with better functionality. For example, they might be able to market the engineering tools for the products and offer the availability of the 3D geometrical representation of the components. At the same time, they would be able to provide a component library, with the associated information and data of the components. With reference from the software industry, they could also introduce product lines [39] of components. The components would not be sold as standalone units, but rather as a family of related and interacting components. From the machine builders' perspectives, they will be able to offer to the end users the service for remote diagnostics. New agreements (such as for training, maintenance and machine servicing) could be made based upon on this service provision.

Improve Core Competencies

The machine builders and the component suppliers will be able to concentrate and improve their core competencies. As the designer of the components, the component suppliers have the best knowledge about the physical build and capabilities of the components. As such, they would be in the best position to provide the necessary software for the components. The move towards component technology would enable the component suppliers to have a more in-depth focus on their core competencies. Similarly, the machine builder will thus be able to concentrate on the process of machine design, build and test. They will not have to worry about the quality of the components and focus their effort on integrating, implementing and servicing the machines.

8.3.4 People

From the people perspective, all the three stakeholders will need new training and be multi-skills / multi-tasks. The common benefits to the machine builders and end users are earlier training and improve diagnostics capability. The machine builders will benefit from tools with better usability and ability to reduce onsite manpower.

Machine Builders

Better Usability for Tools

The machine builders have already been working with the component suppliers to improve the user interface of the engineering tools that the component suppliers provide. Feedback from the machine builders has shown that the use of a component catalogue and a Microsoft Windows based approach for the engineering tools is what they would like to have. Tools are required that are easy and convenient to use, such as being able to select an item from a window, drag and drop it onto the designed spaces. With the use of component technology, the machine builders would be able to create a library of the components that are used most frequently by the engineers. As mentioned earlier, much of the previous machine functions are repeated in new applications and the creation of a library for these control functions would help them to reuse these functions with greater ease. This library of control logic can either be created by the machine builder or it can be supplied by the component suppliers.

Reduce Onsite Manpower

At the present time, the machine builders deploy their engineers when they deliver the machine and the engineers stay on site for a few months or up to a year to make sure that the machines are in good working order. With the availability of remote diagnostics, the engineers might be able to shorten their period of stay, and continue to monitor the machine by accessing it remotely.

Common Issues

Training and Multi-skills / Multi-tasking

For all the three stakeholders, each would require training in designing, implementing and maintaining a CB system. They would need to acquire new knowledge and skills and understand the philosophy behind a CB system and thus how to design and implement the CB architectures that are suitable for reuse and maintenance. For the machine builders, it is likely that their engineers will have to be multi-skilled in order to adopt the CBA. For example, with the introduction of sub-assembly and sub-commissioning, the different engineers might need to have some mechanical, control or commissioning knowledge to enable them to perform the sub-commissioning. For the component suppliers, with the increasing job responsibilities, they will need to learn to implement the software components and the electrical design of the components. If they are to provide 3D models of the components for machine builders, they would need to be conversant in virtual reality modelling languages (VRML). Initially, the investment of resources will be high but when it has been fully assimilated into the organisation, the investment will decrease gradually.

Machine builders and End user

Earlier Start for Training

The virtual representation of the machine can be used for training the operators. The machine builders can provide the end users with the machine model and the associated HMI so that the operators can start their basic training on the operation of the new machines. This would enable end user to bring forward their date for "Job 1" (see section 2.2.2). The machine model would also enable the end user's maintenance engineers to understand the operation of the machine and learn about it at an earlier date, rather than at the last minute when the machine has been delivered and installed on site.

Improved Diagnostic Capability

The difficulties of diagnosing errors on the machine through telephone conversations had been raised by the machine builders and shown in the scenario testing (see section 7.4.1). Often, when they send their engineers on site, the engineers do not have a detailed picture of what is happening on the machine. The availability of machine process simulation using 3D model and by making it available remotely, enables the machine builder to assess the problem before they go onsite. They would have knowledge of the problems and be ready to tackle them straight away.

8.4 Summary

Quantitative and qualitative analysis from the case studies have shown that the CBA offers several potential benefits to the stakeholders in the automotive industry but there are several issues associated with the implementation. These benefits and issues could be discussed from four different aspects: product, process, business and people. The different perspectives of the stakeholders have been taken into consideration in the above discussion. Some of the benefits and issues are of concern to all of the stakeholders, while certain benefits that are enjoyed by an individual stakeholder might be at the expense of other stakeholders. This would require the industry to come to an agreement on a solution that is acceptable to all parties.

Chapter 9

Discussion and Conclusion

9.1 Research Review

The project work on implementation of Component-based approach (CBA) for production machinery in the automotive domain (i.e. CCG projects) at the MSI Research Institute, Loughborough University, has been used as the case study in this research. The primary objectives of the research were to i) identify the impact of the implementation of the CBA in the automotive domain, focusing on the business and the human aspects, and ii) identify and develop a suitable approach for evaluating the CBA.

A strategy has been formulated to evaluate the impact of CBA. The strategy consists of evaluation planning, evaluation design, data collection methods and data analysis. The current evaluation design and data collection methods have been reviewed, forming the basis for the knowledge elicitation methods adopted for this thesis. Additionally, this defined the need to use different data collection methods for both quantitative and qualitative data. Scenario testing has been used in the research to understand the interaction between the user and the engineering tools developed by the CCG projects. A survey was carried out using a questionnaire to understand the stakeholders' view on the CCG projects. The evaluation strategy has also identified the need to adopt the formal enterprise modelling approach for data representation and analysis and integration with dynamic modelling to enable "what if" analysis to be performed. A suitable framework, consisting of four views i.e. product, process, business and people, from the stakeholders' perspectives has been adopted for analysing the impact of the CBA.

The research has concentrated mainly on i) developing an approach to evaluate the CBA and ii) identifying the issues associated with its implementation. The main results of the case studies (Chapter 7) have been presented in Chapter 8 of this thesis.

9.2 Critique of Research

This research has defined an approach for evaluating the implementation of the CBA in the automotive domain. The following sections will discuss the major aspects of the evaluation approach which includes: the approach itself, stakeholders involvement, knowledge elicitation and data representation.

9.2.1 Evaluation Approach

From the literature review (see Chapter 2 and Chapter 3), it was found that there is a lack of a simple and systematic approach to i) evaluate, ii) represent data formally and iii) illustrate the interaction of activities. This research has adopted a mixed method approach in which different evaluation designs (i.e. testing, survey, case studies) and data collection methods (interviews, analysis of documents, questionnaires and scenario testing) have been used. This approach has enabled the author to focus at different aspects of the research, i.e. testing for evaluating a system prior to actual implementation, survey for gathering feedback and case studies for in-depth understanding of the relationships, experiences or processes of a particular stakeholders. In addition, a formal method for data representation has been proposed in this research. Without formal data representation, it would not be possible to appreciate fully the complexity of the lifecycle of machine design and build (section 6.2.4 and Chapter 7) and the processes of implementing a CB approach (section 6.3).

Although the evaluation approach has only been applied in the CCG projects, it can be applied in any research which requires the evaluation of the application of new technology within any domains, be it manufacturing production, financial or services. Interviews, analysis of documents, questionnaires are a set of general methods suitable for knowledge elicitation in any domain. The way these methods are conducted might be different, for example, interviews can be carried out through face-to-face interview, telephone or correspondence. Information elicitation requires documentation. Modelling enables the documented information to be represented formally and graphically. The model enables the abstraction of the essential information and the presentation in a structure that is easy to understand.

9.2.2 Stakeholders Involvement

This research has focused on the machine builders as they are the main player in the process of machine design and build. Comparatively less attention has been given to the end user and the component suppliers. The main attention has been on the main processes of the engineering aspects of machine design and build, other processes such as financial, purchasing, component suppliers work processes, end user's work processes have not been investigated in full detail.

9.2.3 Knowledge Elicitation

This research has adopted interviews, analysis of documents, questionnaires and scenario testing. From the research, it has been found that interviews and analysis of documents were effective in collecting data from the stakeholders. Interviews are effective in eliciting information directly from the stakeholders. However, it is very time consuming. The experience is one interview could easily take two hours and only cover a few topics. An alternative method for eliciting information from the stakeholders is through the use of questionnaires with open-ended questions (see section 2.4.5). The questionnaire would be sent to the interviewee, who would provide written answers. The advantages of using questionnaires, in this case, are that it would be less time-consuming, the opinions of several people could be determined using one questionnaire and different topics could be included. However, it does not provide the interviewer with the opportunity to clarify any doubts immediately and does not allow the interviewer to build up a good rapport with the interviewee to enable effective communication. Therefore, interviews are still preferred as the method to understand fully i) the working practices and culture, and ii) working processes of the stakeholders.

In comparison, analysis of documents has not been as effective in understanding the lifecycle of machine design and build. This is mainly because not all the machine builders keep detailed documentation of the activities in their organisation. In this research, Cross Hüller was the only machine builder who was able to provide good documentation of their engineering processes. This has enabled a process model to be developed. Other machine builders, (i.e. Lamb Technicon and Krause) do not have similar documentation. Other documentations provided by the stakeholders include a list of components used in the test machines, mechanical drawings, electrical drawings and a list of changes applied throughout typical project and how these changes were managed. In this case, the method of analysis of documents was seen as a supplementary data collection method to interviews. However, analysis of documents has been important in understanding the development of CBA i.e. the advantages, the drivers, enablers, in the software industry. The application of the CBA in the software industry is the most mature with respect to the application of the architecture and technology. By referencing CBA as appreciated by the software industry, the advantages and issues of concerns of implementing CBA in the automotive domain have been identified.

Scenario testing has been most useful in assessing how a CB system would be used. User representatives were used in some of the tests (for reasons see section 5.4.2). Nevertheless, the test using user representatives (who have understanding of the work of actual users) has provided an important basis for assessing a "future system". A prototype system has to be available to carry out the test. It is necessary that observers are present to record the results. In cases where actual testing is unable to be carried out, CIMOSA activity diagrams describing the processes involved in the current process and the future processes have been developed. The description enables comparison to be made of the steps involved in carrying out the activities under the AS-IS condition and in the TO-BE situations. However, wherever feasible, the processes described in the diagrams should be validated by trials or experiments. The results from the scenario test are only indicative of what might occur in real situations; more trials or experiments using an actual machines under real life production situation are required in order to get more detailed results.

The survey that was conducted did not provide the intended result due to the poor response ratio, but was able to return with valuable feedback from the stakeholder. The comments from the end user and the component supplier were especially valuable (Chapter 7 and Appendix B) considering that much of the attention had been on the machine builders. With hindsight, it would probably have been better if the survey was conducted before the scenario testing. This would have enabled feedback from the survey to be incorporated in the testing.

9.2.4 Data Representation

By using enterprise modelling methods, the large amount of data elicited form the stakeholders was organised and represented graphically. Although flow charts could have been used for representing processes, however, the resources required and the interactions involved in the processes could not have been fully captured. There is a lack of graphical constructs in flow charts for representing information, events and resources. Static models of the current lifecycle of machine design and build have been developed using a CIMOSA based modelling approach developed in the MSI Research Institute of Loughborough University (i.e. MMA). The MMA provided a set of constructs that enables the representation of the interaction of the different business processes, activities, resources and information within the enterprise. The approach concentrates mainly on capturing the functional aspects of the enterprise; it has limited coverage on modelling the information, resources and the organisation aspects of the enterprise. However, the focus of this research is on functionality, i.e. understanding how the resources support the activities and functions behind the design and implementation process of the automotive engine production machinery. The set of modelling diagrams has provided the essential constructs for describing the processes, activities, and resources within the enterprise. In some cases, however, documentations is still necessary to supplement the static models, for example, documentation is required to describe the knowledge and skill requirement of the stakeholders.

The MMA has also been used for describing the major processes involved in developing a CB system. Using the model, potential users will be able to appreciate the development effort involved.

9.3 Research Achievement and Weakness

The research achievements, the contributions to knowledge and the weakness in this research will be discussed in the following sections.

9.3.1 Research Achievements

The main achievements of this research work are tabulated in Table 9.1. The achievements that have added to the body of knowledge in the field are as follows:

- The identification of a mixed method approach for evaluating the CBA. Methods have been defined for i) understanding the associated domains of interests, ii) assessing future implementation, iii) representing data and iv) data analysis. This approach can be used to help future researchers and practitioners to determine the impact of implementing new technology.
- The adoption of scenario testing and development of a set of scenarios and associated metrics which are suitable for assessing the interactions between users and the future systems to be implemented.
- The adoption of an enterprise modelling approach for data representation and analysis. This has enabled the complexity of the lifecycle of machine design and build, the interactions between stakeholders and the resources requirements to be represented graphically.
- The development of a structure for dynamic modelling to maintain data consistency during the translation of the static models to dynamic model. A modular representation has been developed to facilitate the reuse of the models. Dynamic modelling has enabled 'what-if' analysis to be carried out.
- The adoption of the enterprise modelling methods for representing the main processes involved in the development of a CBA system. This has enabled the functional benefits of innovations, new technical architectures and approaches to be appreciated readily, which i) provides support for analysis of the CB system, ii) promotes understanding of the CB system, iii) enables a better management of the complexity involved in describing the system and iv) supports system design.
- The evaluation of lifecycle view of the machine design and build process to assess the impact of adopting CBA in the automotive sector. This enables the stakeholders to appreciate the impact CBA would have from design, build, testing to maintenance.
- The adoption of a framework which enable the discussion of the impact of implementing CBA from different stakeholders' perspectives and different system development perspectives. The framework has four viewpoints: product, process, business and people, and is viewed from the component supplier, machine builder and end user's perspectives.

- The identification of issues which are of concerned and perceived benefits which will be enjoyed by the stakeholders, component suppliers, machine builders and end users. The discussion helps to bring attention to the disparate perspectives on CBA development, particularly on the business and human aspects. It is hope that through the identification of these issues, bottlenecks that are likely to occur as CBA progresses could be prevented.

9.3.2 Contribution of Knowledge

The contribution of knowledge from this thesis are summarised as followed:

- Detailed understanding of the activities, requirements, issues of concerns in the lifecycle of machine design and build in the automotive sector
- The definition of a set of evaluation methods which is simple and systematic for evaluating new technology
- The application of enterprise modelling methods to supplement classical evaluation methods for data representation and analysis, which reduces complexity, rationalises and secures information flows
- The formulation of a modular structure for dynamic modelling which is reusable and maintains data consistency between the static model and dynamic model.
- The application of enterprise modelling technique to represent the development of a system so as to improve clarity and understanding
- The identification of the benefits and issues of concerns that a CB based control system will bring to the different stakeholders in the automotive industry.

9.3.3 Research Weakness

The weaknesses of this research may prove to be:

- o Different combinations of data collection methods in the evaluation approach has not been investigated due to the size and scope of the research. In addition, the approach has only been tested in the automotive domain. Testing in other domains would help to validate its effectiveness.
- o CBA has not been implemented in a real life production machinery to allow direct comparison with current systems. Although the CBA has been implemented successfully on the industrial collaborator's prototype machine and assessed by the collaborators, it has not been subjected to actual production activities. This inevitably restricted the ability to collect real life production data to confirm CBA's practicality.
- o The use of representatives instead of the actual users. Although user representatives are sufficient for testing, the industrial players will be able to provide invaluable feedback and information from the industry's perspectives on how the system is being used, and how a future system could affect them.
- o The survey sample size was small due to the small number of stakeholders who are aware of the project which inevitably lead to a low response rate.

9.4 Recommended Future Works

There are two areas recommended for future work, one from the evaluation perspective and the other from enterprise perspective.

The evaluation approach developed in this research can be applied to other research work. This will enable the analysis of the effectiveness of the approach. This research has outlined the steps required to designing, planning and conducting the evaluation. There maybe changes to the data collection methods used but the process of carrying out the evaluation is expected to be the same.

From the enterprise perspectives, there are three aspects in an enterprise: strategic, tactical and operational [105]. Strategic refers to the identification of objectives and plans of the enterprise; tactical aspect addresses the means of realising defined goals of the organisation; operational aspects constitutes the set of activities that is being carried out to realise the goals by actually utilising the capabilities, methods and techniques developed in the tactical aspect. To implement CBA successfully in the organisation, strategic planning is required. The switch from current practice to CBA will be a long term investment and it requires comprehensive planning to ensure technical, manufacturing and business success [178]. Strategic planning requires a long term comprehensive view of both business and technology issues [130]. Organisation structure needs to be reviewed and process risk need to be assessed. The plan will provide a top down view of the vision and direction for the organisation, identifying the different phases of changes the organisation has to undergo, review the capability of the organisation and the financial investment requirement.

At the tactical level, enterprise implementing CBA has to look into the organisation changes that are required. Companies need to re-think how their inter and intra-organisation processes will be impacted as a result of CBA investment. These organisational changes would include changes to organisation culture, structure and processes. Important issues to the organisation would be how to maintain employee's morale, psychological effect of changes to employee, internal political, effective communication of changes, acceptance to changes etc; these issues need to be studied carefully so as to introduce changes to the organisation smoothly [179].

Investigation is required to establish a new performance measurement systems for the CBA. A set of criteria is required to measure the performance of CBA such as rate of maintenance, failure rate and errors diagnostic capability. This set of metrics should ideally include the tangible and intangible benefits of CBA. This would enable the organisation to assess its own competitive position and to check on the accomplishment of the organisation's objects or goals [180].

Research Objectives	Research Achievements	
	Achievements	Not Achieved
To use CCG as a case study for evaluating the implementation of CBA in automotive domain	The CCG has been used as a case study for understanding the implementation of CBA in the automotive sector	---
To develop a strategy for evaluating CBA	A structured methods for evaluation has been developed consisting of evaluation design, data collection methods and data analysis.	---
To identify methods for evaluation	A mixed method approach has been adopted; the evaluation design is based on the use of case studies, testing and survey. Data collection methods identified include of interviews, analysis of documents, scenario testing and questionnaire (i.e. survey).	The survey has not been able to gather feedback from all the targeted stakeholders
To identify a suitable method for testing future implementation	Scenario testing has been adopted and is used for assessing the interactions between the users and a future system. A set of metrics has been identified in each of the scenarios.	Not all the scenarios created have been fully tested due to the limitation of resources and time.

Table 9.1 Research Achievements

Research Objectives	Research Achievements	
	Achievements	Not Achieved
To use enterprise modelling method for data representation and analysis	A CIMOSA based modelling approach has been used for capturing and representing the data collected. The static model created formed the basis for dynamic modelling.	The effect of changes in cost was not fully simulated through dynamic modelling due to the unavailability of the data.
	A structure for dynamic modelling has been established to ensure data consistency between the static model and dynamic model. The structure is modular and enables the reuse of the model. The dynamic modelling proof to be useful in carrying out "what-if" analysis especially with regards to time resources.	--
	To use the CIMOSA based modelling approach to identify the processes involved in developing a CB system.	--

Table 9.1 Research Achievements (continue from previous page)

Research Objectives	Research Achievements	
	Achievements	Not Achieved
To discuss the impact of implementing CBA with respect to stakeholders	The evaluation of lifecycle view of the machine design and build process to assess the impact of adopting CBA in the automotive sector, enabling the stakeholders to appreciate the impact CBA would have from design, build, testing to maintenance.	--
	A suitable framework consisting of product, process, business and people has been adopted and used in this research for discussing the impact of implementing CBA. The framework has been used for discussing the benefits and the issues concerned.	---
	A set of benefits and issues related to the component suppliers with the implementation of CBA has been identified and discussed.	The work process and the stakeholders within the component supplier need to be studied in order to fully understand and verify the impact CBA implementation.

Table 9.1 Research Achievements (continue from previous page)

Research Objectives	Research Achievements	
	Achievements	Not Achieved
	A good understanding of the work process of the machine builders, especially in transfer line, had been achieved and benefits and issues concerned to the machine builders were identified. A set of scenario testing was conducted with the machine builders; both static and dynamic modelling were carried out. Comparison of the AS-IS and TO-BE process has been made.	Detailed information for the assembly line machine builder has not been available to facilitate dynamic modelling.
	The requirements of the end users were considered for scenario testing. A set of benefits and issues related to the end users with the implementation of CBA has been identified and discussed.	It will be necessary to capture the work processes within the end user and would be helpful if tests could be conducted with the end user to fully appreciate the impact CBA would bring to them.

Table 9.1 Research Achievements (continue from previous page)

Reference

[1] Anonymous, "Challenges Facing the Global Automotive Industry," in *Consumer and Engineered Products*, vol. 1: Booz, Allen & Hamilton, 1999.

[2] Anonymous, "Review of Automotive Assistance," Australia Productivity Commission, Position Paper 2002.

[3] Y. Kakino, "New Trend of Production in Japanese Car Industries," presented at Proceedings of the 2000 Japan-USA Flexible Automation Conference, Ann Arbor, Michigan, US, 2000.

[4] H. K. Toenshoff and A. Schnuelle, "High Productive And Reconfigurable Manufacturing Systems (HIPARMS)," presented at IST 99 Helsinki : Information Society Technology Conference 1999, Helsinki, 1999.

[5] Anon, "Assessment and Implementation of a Component-Based Paradigm for Agile Automation," GR/M43586, EPSRC, Innovative Manufacturing Initiative, Project Proposal for Grant Application 1998.

[6] P. J. Rogers and G. Hough, "Improving the Effectivenss of Evaluations: Making the Link to Organisational Theory," *Evaluation and Program Planning*, vol. 18, pp. 321-332, 1995.

[7] F. L. Hoffmann, E. Leckman, N. Russo, and L. Knauf, "In it for the Long Haul: the Integration of outcomes assessment, clinical services, and management decision-making," *Evaluation and Program Planning*, vol. 22, pp. 211-219, 1999.

[8] M. H. Ong, "Evaluating the Impact of Adopting the Component-Based System within the Automotive Domain," Loughborough University, Loughborough, United Kingdom, Thesis submitted for the award of Doctor of Philosophy 2004.

[9] S. Mari, "Automotive Industry," International Motor Vehicle Program 2000.

[10] M. Ramage, "The Learning Way: Evaluating Co-operative Systems," Department of Computing, Lancaster University, Lancaster, United Kingdom, Thesis submitted for the award of Doctor of Philosophy 1999.

[11] Anonymous, "Industry as a Partner for Sustainable Development : Automotive," International Automobile Industry and United Nations Environment Programme 2002.

[12] Anonymous, "Component-Based Design and Integration Platforms : A Roadmap for Advanced Real-Time Systems," ARTIST Consortium, Information Society Technologies Project 2003.

[13] L. Fabry, "Trends in European Foreign Automotive Investments," Business Briefings Ltd 2003.

[14] P. C. Clements, "From subroutines to subsystems: Component-Based Software Development," *The American Programmer*, vol. 8, 1995.

[15] K. Shimokawa, "Reorganisation of the Global Automobile Industry and Structural Change of the Automobile Component Industry," TokaiGakuen University, International Motor Vehicle Programme Working Papers 2000-2001, Massachusetts Institute of Technology 2000.

[16] M. Freyssenet and Y. Lung, "Between Globalisation and Regionalisation: What is the Future of the Automobile Industry?," GERPISA International Network 1999.

[17] A. J. M. Monteiro, "Production Cost Modelling for the Automotive Industry," in *Instituto Superior Tecnico*: Universidade Technica De Lisboa, 2001.

[18] F. Pilorusso, "Finding a Place in the Automotive Supplier Hierarchy in the Year 2000 and Beyond," International Motor Vehicle Program 1998.

[19] Anonymous, "Automotive Industry Online - Executive Summary," eMarketer, Market Report 2003.

[20] Anonymous, "Mergers, Takeovers and Product Differentiation, http://www.bized.ac.uk/educators/16-19/activity/vce_business/2003_4/230204.htm," Institute for Learning and Research Technology (ILRT), University of Bristol.

[21] T. Sturgeon, "Globalisation and Jobs in the Automotive Industry," International Motor Vehicle Program, Massachusetts Institute of Technology, October 1997.

[22] P. T. Kidd, "Revolutionising New Product Development: A blueprint for success in the global automotive industry," FT Automotive Publishing, Management Report 1997.

[23] D. F. Jambor, "Supplier Integration," presented at Automotive Manufacturing Autotech '97, Birmingham, UK, 1997.

[24] C. Chandler, "Globalisation: The Automotive Industry's Quest for a "World Car"," in *http://globaledge.msu.edu/KnowledgeRoom/FeaturedInsights/*.

[25] Anonymous, "Mission to the USA Automotive Industry," Foresight Vehicle - Advanced Materials and Structures Thematic Group (FASMAT) 1999.

[26] A. Camuffo, "Rolling out a "World Car" : Globalisation, Outsourcing and Modularity in the auto Industry," Department of Business Economic and Management, Ca' Foscari University of Venice, Italy, Venice, International Motor Vehicle Programme Working Papers 2000-2001, Massachusetts Institute of Technology.

[27] S. Fixson and S. Mari, "Modularity in Product Architecture: Will the Auto Industry Follow the Computer Industry?," International Motor Vehicle Programme, Paper presented at Fall Meeting 2001 2001.

[28] M. Brylawski, "Uncommon Knowledge: Automotive Platform Sharing's Potential Impact on Advanced Technologies," presented at First International Society for the Advancement of Material and Process Engineering, automotive Conference, Detroit, Michigan, 1999.

[29] Anonymous, "Ford selects Tupy and SinterCast for New Engine Programme," Sintercast AB, Press Release from Sintercast 2001.

[30] Anonymous, "Dictionary of Automotive Terms."

[31] R. P. Monfared, A. A. West, R. Harrison, and R. H. Weston, "An Implementation of the business process modelling approach in the automotive industry," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 216, pp. 1413-1428, 2002.

[32] R. Harrison, S. M. Lee, and A. A. West, "Component Based Distributed Control System for Automotive Manufacturing Machinery Developed Under the Foresight Vehicle Programme," presented at SAE Congress 2002, 2002.

[33] R. Harrison and A. A. West, "Component Based Paradigm for the Design and Implementation of Control Systems in Electronics Manufacturing Machinery," *Journal of Electronics Manufacturing*, vol. 10, pp. 1-17, 2000.

[34] R. Harrison, A. A. West, and R. P. Monfared, "Distributed Engineering of Manufacturing Machines," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 215, pp. 217-231, 2001.

[35] M. H. Ong, "Case Study - Engine Development Project," Loughborough University, Project Report 2002.

[36] C. D. Wright and K. Case, "Emulation of Modular Manufacturing Machines using CAD Modelling," *Mechatronics*, vol. 4, pp. 713-735, 1995.

[37] A. A. West, R. Harrison, C. D. Wright, and A. J. Carrott, "The visualisation of Control Logic and Physical Machine Elements within an Integrated Machine Design and Control Environment," *Mechatronics*, vol. 10, pp. 668-698, 2000.

[38] M. H. Ong, "The machine design and building activities in Krause," MSI Research Institute, Loughborough University, Internal Report July 2001.

[39] L. Bass, C. Buhman, S. Comella-Dorda, F. Long, J. Robert, R. Seacord, and K. Wallnau, "Market assessment of Component-Based software engineering," Software Engineering Institute, Carnegie Mellon University May 2000.

[40] F. P. J. Brooks, "No Silver Bullet : Essence and accidents of software engineering," *Computer*, vol. 20, pp. 10-19, 1987.

[41] P. Brereton and D. Budgen, "Component-Based Systems: a classification of issues," *Computer*, pp. 54-62, 2000.

[42] Anon, "Component Software Development / COTS Integration," Software Engineering Institute, Carnegie Mellon University.

[43] T. D. Meijler and O. Nierstrasz, "Beyond Objects: Components," in *Cooperative Information Systems: Current Trends and Directions*, G. Schlageter, Ed.: Academic Press, 1997.

[44] R. Veryard, "Software Componentry, http://www.users.globalnet.co.uk/~rxv/CBDmain/," Veryard-Projects.

[45] K. McInnis, "Component-based Development : The Concepts, Technology and Methodology," Castek Software Factory Inc. 2000.

[46] H. Washizaki, "A study on realisation of component-based software development technology," Graduate School of Science and Engineering, Waseda University, Tokyo, Japan, Thesis submitted for the award of Doctor of Philosophy 2003.

[47] L. Deri, "A Component-based Architecture for Open, Independently Extensible Distributed Systems," University of Berne, Switzerland, 1997.

[48] M. Sparling, "Lessons learned through six years of component-based development," *Communications of the ACM*, vol. 43, pp. 47-53, 2000.

[49] whatis.com, "searchNetworking.com Definitions."

[50] J. C. Butler, D. R. Mayo, and J. Weiler, "Succeeding with Component-Based Architecture in e-Government," Federal Enterprise Architecture Program Management Office, White Paper 2003.

[51] M. Aoyama, "New Age of Software Development : How Component-Based Software Engineering Changes the Way of Software Development," presented at 1998 International Workshop on Component-Based Software Engineering, 1998.

[52] M. H. Ong, R. P. Monfared, S. M. Lee, W. A.A., and R. Harrison, "Evaluating the Implementation of the Component-Based System in the Automotive Sector," presented at Proceedings of 7th Biennial ASME Conference: Engineering Systems Design and Analysis (ESDA 2004), Manchester, United Kingdom, 2004.

[53] M. Q. Patton, *Creative Evaluation*: Sage Publications, 1981.

[54] Anon, "User-Friendly Handbook for Project Evaluation, http://www.ehr.nsf.gov/rec/programs/evaluation/handbook/," The Division of Research, Evaluation and Communication.

[55] D. Russ-eft, R. Atwood, and T. Egherman, "Use and Non-use of Evaluation Results: Case Study of Environmental Influences in the Private Sector," *American Journal of Evaluation*, vol. 23, pp. 19-31, 2002.

[56] H. Preskill and V. Caracelli, "Current and Developing Conceptions of Use : Evaluation Use TIG Survey Results," *Evaluation Practice*, vol. 18, pp. 209-225, 1997.

[57] L. M. Shulha and J. B. Cousins, "Evaluation Use: Theory, Research, and Practice since 1986," *Evaluation Practice*, vol. 18, pp. 195-208, 1997.

[58] M. Scriven, *Evaluation Thesaurus*: Edgepress, 1991.

[59] P. H. Rossi and H. E. Freeman, *Evaluation: A Systematic Approach*, 2nd ed. London: Sage, 1982.

[60] E. G. Guba and Y. S. Lincoln, *Fourth Generation Evaluation*: SAGE Publications, 1989.

[61] M. Q. Patton, "A World Larger than Formative and Summative," *Evaluation Practice*, vol. 17, pp. 131-144, 1996.

[62] M. Q. Patton, *Utilization-Focused Evaluation*: Sage Publications, 1997.

[63] Anon, "Program Evaluation Methods : Measurement and Attribution of Program Results," Treasury Board of Canada, Secretariat 3rd Edition, 1998.

[64] M. Denscombe, *The Good Research Guide for Small Scale Social Research Projects*: Open University Press, 1998.

[65] N. L. Sproull, *Handbook of Research Methods: A Guide for Practitioners and Students in the Social Sciences*, 2nd ed: Scarecrow Press, 1995.

[66] C. Frankfor-Nachmias and D. Nachmias, *Research Methods in the Social Sciences*, 4th ed: Edward Arnold, 1992.

[67] W. L. Neuman, *Social Research Methods : Qualitative and Quantitative Approaches*, 2nd ed: Allyn and Bacon, 1994.

[68] W. M. Trochim, "Research Methods Knowledge Base, http://trochim.human.cornell.edu/kb/."

[69] C. Robson, *Real World Research: a resource for social scientists and practitioner-researchers*: Blackwell Publishers, 1996.

[70] Anon, "Research Methods and Statistics Resources, http://www.georgetown.edu/departments/psychology/researchmethods/researchanddesign/quantitative.htm," Department of Psychology, Georgetown University.

[71] S. M. Lee, "A Component-Based Distributed Control Paradigm for Manufacturing Automation Systems," Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, United Kingdom, Thesis submitted for the award of Doctor of Philosophy 2004.

[72] Anon, "Teaching and Learning: Research methodologies, http://www.ucd.ie/teaching/res/non.htm," Centre for Teaching and Learning, University College Dublin.

[73] J. B. Williamson, D. A. Karp, J. R. Dalphin, and P. S. Gray, *The Research Craft: An Introduction to Social Research Methods*, 2nd ed: Little, Brown and Company, 1977.

[74] M. Mitchell and J. Jolley, *Research Design Explained*: Holt, Rinehart and Winston, Inc., 1988.

[75] B. S. Phillips, *Social Research: Strategy and Tactics*. New York: The Macmillan Company, 1971.

[76] R. K. Yin, *Case Study Research: Design and Methods*. London: Sage, 1984.

[77] D. De-Vaus, *Research Design in Social Research*: SAGE Publications Ltd, 2001.

[78] J. Rowley, "Using Case Studies in Research," *Management Research News*, vol. 25, pp. 16-27, 2002.

[79] D. B. Morris, "The Inclusion of Stakeholders in Evaluation : Benefits and Drawbacks," *The Canadian Journal of Program Evaluation*, vol. 17, pp. 49-58, 2002.

[80] P. R. Brandon, "Stakeholder Participation for the Purpose of Helping Ensure Evaluation Validity: Bridging the Gap between Collaborative and Non-Collaborative Evaluations," *American Journal of Evaluation*, vol. 19, pp. 325-337, 1998.

[81] "The Qualitative/Quantitative Debate," Writing Center at Colorado State University, http://writing.colostate.edu/index.cfm.

[82] D. Dswenson, "Sampling and Surveying Handbook, http://www.css.edu/users/dswenson/web/question.htm."

[83] M. Q. Patton, *Qualitative evaluation and research methods*, 2nd ed: SAGE Publications, 1990.

[84] A. Fink, *How to Manage, Analyse, and interpret Survey Data*, 2nd ed: SAGE Publications, 2003.

[85] T. L. Baker, *Doing Social Research*, 2nd ed: McGraw-Hill, 1988.

[86] M. Miles and A. Huberman, *Qualitative Data Analysis*: Thousand Oaks, CA:SAGE, 1994.

[87] K. Garland, "Research Method," http://www.garland.f9.co.uk/RM/protocols.htm.

[88] M. Joppe, "The Research Process, http://www.ryerson.ca/~mjoppe/ResearchProcess/."

[89] C. Robson, *Real World Research : A Resource for Social Scientists and Practitioner-Researchers*, 2nd Edition ed: Blackwell Publishers, 2002.

[90] F. B. Vernadat, *Enterprise Modeling and Integration: Principles and Applications*: Chapman & Hall, 1996.

[91] D. H. Liles and A. R. Presley, "Enterprise modelling within an enterprise engineering framework," presented at Proceedings of the 28th conference on Winter simulation, Coronado, California, United States, 1996.

[92] M. O. Adigun and D. P. Biyela, "Modelling an Enterprise for Re-engineering : A Case Study," presented at Proceedings of the 2003 annual research conference of the South African institute of Computer Scientists and Information Technologists on Enablement through technology, Johannesburg, South Africa, 2003.

[93] K. H. Choi, C. H. Bae, and S. H. Lee, "Behaviour Modelling and Control of Computer Integrated Manufacturing," *International Journal of Computer Integrated Manufacturing*, vol. 16, pp. 128-139, 2003.

[94] K. T. K. Toh, J. A. Harding, and D. Thompson, "A Holonic Approach for the Modelling of Enterprise Functionality and Behaviour," *International Journal of Coomputer Integrated Manufacturing*, vol. 12, pp. 541-558, 1999.

[95] R. Weston, P. Clements, D. N. Carrott, A. Hodgson, and A. A. West, "On the Explicit Modelling of Systems of Human Resources," *International Journal of Production Research*, vol. 39, pp. 185-204, 2001.

[96] J. A. Harding, B. Yu, and K. Popplewell, "Information Modelling : An Integration of Views of a Manufacturing Enterprise," *International Journal of Production Research*, vol. 37, pp. 2777-2792, 1999.

[97] Anonymous, "Scenario Planning - A Strategic Tool," Bureau of Rural Sciences, Department of Agriculture Fisheries and Forestry, Australia.

[98] K. Kosanke, F. Vernadat, and M. Zelm, "CIMOSA: enterprise engineering and integration," *Computers in Industry*, vol. 40, pp. 83-97, 1999.

[99] A. Zwegers, "On Systems Architecting: a study in shop floor control to determine architecting concepts and principles." Eindhoven, The Netherlands: Teschnische Universiteit Eindhoven, 1998, pp. 171.

[100] P. Bernus and L. Nemes, "Enterprise Integration - Engineering Tools for Designing Enterprises," in *Modelling and Methodologies for Enterprise Integration*, L. Nemes, Ed.: Chapman & Hall, 1996, pp. 3-11.

[101] G. Keller and S. Detering, "Process-Oriented modeling and analysis of Business Processes using the R/3 Reference Model," in *Modelling and Methodologies for Enterprise Integration*, L. Nemes, Ed.: Chapman and Hall, 1995.

[102] K. Kosanke and M. Zelm, "CIMOSA Modelling Processes," *Computers in Industry*, vol. 40, pp. 141-153, 1999.

[103] B. Curtis, M. I. Kellner, and J. Over, "Process Modelling," *Communications of the ACM*, vol. 35, pp. 75-89, 1992.

[104] K. Kosanke, "Process Oriented Representation of Modelling Methodologies," in *Modelling and Methodologies for Enterprise Integration*, L. Nemes, Ed.: Chapman and Hall, 1996, pp. 45-55.

[105] K. A. Chatha, "Multi-Process Modelling Approach to Complex Organisation Design," Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, UK, Thesis submitted for the award of Doctor of Philosophy 2003.

[106] G. Bellinger, "Dynamic modelling, http://www.systems-thinking.org/dynmod/dynmod.htm."

[107] A. Greasley, "Using Business-Process Simulation within a Business-Process REengineering Approach," *Business Process Management Journal*, vol. 9, pp. 408-420, 2003.

[108] R. H. Weston, "Workbenches and Reference Models for Enterprise Engineering," in *Handbook of Life Cycle Engineering: Concepts, Models and Technologies*, A. Kusiak, Ed.: Kluwer Academic Publishers, 1998, pp. 597-616.

[109] Anonymous, "Component Objects: Technology Overview," Microsoft Corporation 1993.

[110] T. Williams and G. Rathwell, "Pera Enterprise Integration Web Site, http://www.pera.net."

[111] G. Doumeingts, "GIM, GRAI Integrated Methodology," in *Handbook of Life Cycle Engineering: Concepts, Models and Technologies*, A. Kusiak, Ed.: Kluwer Academic Publishers, 1998, pp. 227-288.

[112] Anonymous, "GIM - GRAI Integrated Methodology, http://www.atb-bremen.de/projects/prosme/Doku/oqim/GRAI.htm."

[113] Anonymous, "Enterprise Reference Architecture, http://www.atb-bremen.de/projects/prosme/Doku/oqim/INDEX.htm," Institute for Applied Systems Technology.

[114] T. J. Williams, "The Purdue Enterprise Reference Architecture and Methodology (PERA)," in *Handbook of Life Cycle Engineering: Concepts, Models and Technologies*, A. Kusiak, Ed.: Kluwer Academic Publishers, 1998, pp. 289-330.

[115] Anonymous, "GERAM : Generalised Enterprise Reference Architecture and Methodology," International Federation for Information Processing, IFIP-IFAC Task Force on Architectures for Enterprise Integration 1999.

[116] SCENIC, "Support Center Network for IT Construction."

[117] Anonymous, "CIM Open System Architecture, http://cimosa.cnt.pl/."

[118] K. Mertins and R. Jochem, "Integrated Enterprise Modelling: Reference Architecture and Methodology," in *Handbook of Life Cycle Engineering: Concepts, Models and Technologies*, A. Kusiak, Ed.: Kluwer Academic Publishers, 1998, pp. 359-395.

[119] M. H. Ong, "Documentation on Lamb Technicon - General," MSI Research Institute, Loughborough University, Project Documentation 2002.

[120] T. Tommila, O. Venta, and K. Koskinen, "Next Generation Industrial Automation - Needs and Opportunities," *Automation Technology Review*, pp. 34-41, 2001.

[121] H. Kopetz, "Component-based design of large distributed real-time systems," *Control Engineering Practice*, vol. 6, pp. 53-60, 1997.

[122] D. M. Dilts, N. P. Boyd, and H. H. Whorms, "The Evolution of Control Architecture for Automated Manufacturing Systems," *Journal of Manufacturing System*, vol. 10, pp. 79-93, 1991.

[123] J. A. T. and C. R. McLean, "A Proposed Hierarchical Control Model for Automated Manufacturing Systems," *Journal of Manufacturing System*, vol. 5, pp. 15-25, 1986.

[124] R. Harrison, J. Pu, R. H. Weston, P. R. Moore, and A. H. Booth, "The Use of Distributed Programmable Actuators for Flexible Assembly," presented at Proceedings of Assembly Automation, Japan, 1989.

[125] R. H. Weston, R. Harrison, and P. R. Moore, "Industrial Robots - Modular Systems," in *Encyclopaedia of Systems and Control*, M. Sing, Ed.: Pergamon, 1988, pp. 2459-2464.

[126] R. Harrison, R. H. Weston, A. H. Booth, and P. R. Moore, "Modular Machines and a New Approach to Machine Control," *Computer Integrated Mechanical Engineering*, 1989.

[127] Anonymous, "Component Based Automation," Siemens AG, 2000.

[128] Anonymous, "Interface for Distributed Automation : Architecture Description and Specification," Interface for Distributed Automation Group Revision 1.0, 2001.

[129] D. Tilbury and P. Kargonekar, "Report of the NSF Workshop : Challenges and Opportunities in Logic Control for Manufacturing Systems," presented at NSF Workshop, University of Michigan, Ann Arbor, 2000.

[130] R. Millen and A. S. Sohal, "Planning Processes for Advanced Manufacturing Technology by Large Americal Manufacturers," *Technovation*, vol. 18, pp. 741-750, 1998.

[131] G. J. Udo and I. C. Ehie, "Advanced Manufacturing Technologies : Determinants of Implementation Success," *International Journal of Operations & Production Management*, vol. 16, pp. 6-26, 1996.

[132] M. H. Small and I. J. Chen, "Economic and Strategic Justification of AMT Inferences from Industrial Practices," *International Journal of Production Economics*, vol. 49, pp. 65-75, 1997.

[133] G. P. White, "A survey and taxonomy of strategy-related performance measures for manufacturing," *international Journal of Operations & Production Management*, vol. 16, pp. 42-61, 1994.

[134] R. P. Monfared, A. A. West, R. Harrison, and D. A. Vera, "Application of Business Process Modelling to Assess the Impact of Process Change," presented at Building the knowledge Economy, Proceedings of eChallenges Conference,e2003, Amsterdam, 2003.

[135] D. A. Vera, "A Component Based Approach to the Design and Implementation of a Virtual Prototyping Environment for Manufacturing Systems," Wolfson School of Mechanical and Manufacturing Department, Loughborough University, Loughborough, United Kingdom, Thesis submitted for the award of Doctor of Philosophy 2004.

[136] D. W. Thomas, A. A. West, R. Harrison, and C. S. McLeod, "A Process Definition Environment for Component Based Manufacturing Machine Control Systems Developed Under the Foresight Vehicle Programme," *Proceedings of the Society of Automotive Engineers World Congress and Exposition, Paper no. 2002-01-0468, Foresight Vehicle Technology: Supply Chain & Manufacturing/Machine Manufacturing Control (Part C&D)*, 2002.

[137] E. W. Mellor, R. Harrison, and A. A. West, "A Component-Based Human Machine Interface System for Automotive Manufacturing Machines," presented at Proceedings of 7th Biennial ASME Conference: Engineering Systems Design and Analysis 2004, Manchester, United Kingdom, 2004.

[138] W. B. Sanders and T. K. Pinhey, *The Conduct of Social Research*: Holt, Rinehart and Winston, 1983.

[139] R. Harrison, S. M. Lee, and A. A. West, "Lifecycle Engineering of Modular Automated Machines," in *2nd IEEE International Conference on Industrial Informatics INDIN'04*, G. Schreck, Ed. Berlin, Germany: Fraunhofer, 2004, pp. 501-506.

[140] S. M. Lee, R. Harrison, and A. A. West, "The Development and Implementation of Manufacturing Automation Systems using Smart Mechatronics Components," presented at Mechàntronics and Robotics Conference '04, Aachen-Germany, 2004.

[141] D. A. Vera, A. A. West, R. Harrison, and D. W. Thomas, "Virtual Visualisation and Prototyping Environment for Component-Based Production Machinery," presented at NAMRC XXXI, Hamilton, Ontario, Canada, 2002.

[142] S. M. Lee, R. Harrison, and A. A. West, "A Component-based Distributed Control System for Assembly Automation," presented at 2nd IEEE International Conference on Industrial Informatics INDIN'04, Berlin, Germany, 2004.

[143] D. McKenzie, "Getting Started with VRML, http://www.vrmlsite.com/aug96/spotlight/intro/intro.html."

[144] Anon, "COmmon Model for PArtNers in automatION (COMPANION) : An integrated engineering environment offering greater agility in the lifecycle of automotive manufacturing machinery," GR/M54042, EPSRC, Systems Integration Programme, Project Proposal for Grant Application 1999 1999.

[145] C. H. Weiss, *Evaluation research :methods for assessing program effectiveness*: Prentice-Hall, U.S., 1972.

[146] J. Riley, *Getting the Most from Your Data*: Technical and Educational Services Ltd, 1990.

[147] D. Philpott, "An Investigation of the Socio-Technical Implications of a Computer Based Walk Optimisation System," in *Department of Human Sciences*. Loughborough, United Kingdom: Loughborough University, 1998.

[148] J. M. Carroll, "Scenario-Based Design," in *Handbook of Human-Computer Interaction*, P. V. Prabhu, Ed., 2nd Editon ed: Elsevier, 1997.

[149] J. M. Carroll, "Five reasons for scenario-based design," *Interacting with Computers*, vol. 13, pp. 43-60, 2000.

[150] J. Nielsen, "Scenarios in discount usability engineering," in *Scenario-based design: envisioning work and technology in system*, J. M. Carroll, Ed.: John Wiley & Sons, Inc., 1995, pp. 59-83.

[151] J. Karat, "Scenario Use in the Design of a Speech Recognition System," in *Scenario-Based Design: Envisioning Work and Technology in System Development*, J. M. Carroll, Ed.: John Wiley & Sons, 1995.

[152] A. Sutcliffe, "Symbiosis and Synergy? Scenarios, Task Analysis and Reuse of HCI Knowledge," *Interacting with Computers*, vol. 15, pp. 245-263, 2003.

[153] J. M. Carroll, "Making Use is More Than a Matter of Task Analysis," *Interacting with Computers*, vol. 14, pp. 619-627, 2002.

[154] L. Liu and E. Yu, "Designing Information Systems in Social Context: a Goal and Scenario Modelling Approach," *Information Systems*, vol. 29, pp. 187-203, 2004.

[155] A. MacLean and D. McKerlie, "Design Space Analysis and Use Representations," in *Scenario-Based Design: Envisioning Work and Technology in System Development*, J. M. Carroll, Ed.: John Wiley & Sons, 1995.

[156] M. Hertzum, "Making Use of Scenarios : A Field Study of Conceptual design," *International Journal of Human-Computer Studies*, vol. 58, pp. 215-239, 2003.

[157] K. Eason and W. Olphert, "Early Evaluation of the Organisational Implications of CSCW Systems," in *CSCW Requirements and Evaluation*, P. J. Thomas, Ed. London, UK: Springer, 1996, pp. 75-89.

[158] R. L. Mack, "Discussion: Scenarios as Engines of Design," in *Scenario-Based Design: Envisioning Work and Technology in System Development*, J. M. Carroll, Ed.: John Wiley & Sons, 1995.

[159] J. M. Carroll and M. B. Rosson, "Getting Around the Task-Artifact Cycle: How to Make Claims and Design by Scenario," *ACM Transactions on Information Systems*, vol. 10, No. 2, pp. 181-212, 1992.

[160] M. Dong, "Process Modeling, Performance Analysis and Configuration Simulation in Integrated Supply Chain Network Design," Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Virginia Thesis submitted for award of Doctor of Philosophy, 2001.

[161] A. J. R. Zwegers and T. A. G. Gransier, "Managing re-engineering with the CIMOSA architectural framework," *Computers in Industry*, vol. 27, pp. 143-153, 1995.

[162] E. Pierard, "Reference architecture for car assembly monitoring," *Computers in Industry*, vol. 27, pp. 203-213, 1995.

[163] R. P. Monfared, "A component-based approach to design and construction of change capable manufacturing cell control systems," Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, UK, Thesis submitted for the award of Doctor of Philosophy 2000.

[164] M. Zelm, F. B. Vernadat, and K. Kosanke, "The CIMOSA business modelling process," *Computers in Industry*, vol. 27, pp. 123-142, 1995.

[165] K. Kosanke, "CIMOSA - Overview and status," *Computers in Industry*, vol. 27, pp. 101-109, 1995.

[166] M. W. C. Aguiar, "An Approach to Enacting Business Process Models in Support of the Lifecycle of the Integrated Manufacturing Systems." PhD Thesis, Manufacturing Engineering Department, Loughborough University, Loughborough, U.K., 1995.

[167] Microsoft, "Microsoft PowerPoint, http://office.microsoft.com."

[168] Microsoft, "Visio Programming," 2002 ed: Microsoft Press.

[169] G. M. Giaglis, "Transformation of Static Process Models into Dynamic Simulations: Issues and Considerations," in *Process Modelling*, A. Nethe, Ed.: Springer-Verlag, BErlin, 1999, pp. 161-176.

[170] Anonymous, "iThink Analyst, http://www.hps-inc.com," Version 6.0.1 ed: High Performance Systems.

[171] D. Aronson, "Overview of Systems Thinking."

[172] G. Bellinger, "Mental Model Musings - Systems Thinking : An Operational Perspective of the Universe."

[173] Anonymous, *iThink Technical Documentation*: High Performances System Inc, 1996.

[174] C. Szyperski, *Component Software: Beyong Object-Oriented Programming.* New York: ACM Press and Addison-Wesley, 1998.

[175] I. Crnkovic, M. Larsson, and F. Luders, "State of the Practice: Component-based Software Engineering Course," presented at Proceedings of Workshop ICSE 2000 Conference, 3rd International Workshop, 2000.

[176] M. R. Lucas, "Understanding and Assessing Logic Control Design Methodologies," University of Michigan, Ann Arbor, Michigan, United States of America, Thesis submitted for the degree of PhD 2003.

[177] Anonymous, "Simulation Helps Increase Reliability of Project Proposals and System Performance," Delmia Solutions, Company Press Release 2000.

[178] A. Estathiades, S. Tassou, and A. Antoniou, "Strategic planning, transfer and implementation of Advanced Manufacturing Technologies (AMT). Development of an Integrated Process Plan," *Technovation*, vol. 22, pp. 201-212, 2002.

[179] M. McGreevy, "The changing nature of work," *Industrial and Commercial Training*, vol. Vol. 35, pp. 191-195, 2003.

[180] A. DeToni and S. Tonchia, "Performance measurement systems: Models, characteristic and measures," *International Journal of Operations & Production Management*, vol. 21, pp. 46-70, 2001.

Appendix A

CIMOSA Modelling Diagrams for CCG Projects

CIMOSA modelling diagrams were drawn to represent the activities for creating a Component Based system under the CCG projects. These diagrams explain the processes and the resources involved; for the technical details, readers are to refer to the references: Lee [71] for details on the system architecture and component architecture, Vera [135, 141] for VRML model creation and Mellor [137] for HMI creation.

The domain is identified as the CB system as shown in Figure A.2. Six CIMOSA domains had been identified as the DP; they are 'Create machine DP1', 'VRML 3D Model DP2', 'Create Component DP3', 'HMI DP4', 'Machine Simulation DP5', 'Machine Monitoring DP6'. The order of these DPs is not in sequential order. The processes are independent of each other but there are interactions among the DPs. The other three non-CIMOSA domains are: 'Common Database', 'Control Environment' and 'Process Definition Environment'.

Create Machine

Figure A.3 shows the structure diagram for 'Create Machine DP1'. Two BPs have been identified: "Machine ConfigurationBP1.1" and 'Test and Debug 1.11'. 'Machine Configuration' describes the processes of creating a machine using the engineering tool, known as the Process Definition Environment (PDE) developed under the CCG projects. The machine creation starts with creating a system in a Tree Hierarchy diagram. The next level of the system is the subsystem. The subsystem could be populated using two views, the component views (BP1.3) or by using the modules view (BP1.14). Using the component views, components are selected from a component library in the PDE. When all the components within the subsystem have been selected, interlocks (which states the conditions for the element to change its state) are being defined (EA1132). Using the modules view, modules are added to the subsystem instead of individual components (EA1151). A module consists of components and their geometrical model representation. Configuring the machine using modules would mean building the machine virtually. Link points are defined so that the modules are joined at specific position or location (EA1152). Again, interlocks had to be defined so as to coordinate the sequential behaviour of the components (EA1153).

The BP 'Test & Debug BP1.2' describes the process of testing and debugging of the configured machine using the PDE. There are two ways of simulating the machine logic, (i.e. two sub-BPs); this could be done using the states transition diagram view (BP122) or using the 3D model view (BP123). Under the states transition diagram view, the machine to be simulated is selected (EA1221). The mode of simulation is selected (BP1221), which is either manual simulation (BP12211) or automatic simulation (BP12212), as shown in Figure A.1. For manual simulation, the machine parts had to be loaded (EA122111) and the sensors are toggled (EA122112) to trigger off the machine sequences. Using automatic simulation, likewise the machine has to be loaded (EA122121). The simulation is started by clicking on a button provided on the PDE window (EA122122). Using the 3D model view (BP123), the VRML 3D model of the machine has to be loaded (EA1231). The logical parts of the machine are loaded to the model (EA1232) and simulation starts by clicking on the button provided by the PDE.

Figure A.4 shows the activity diagram for machine creation. It captures the different resources, information exchanges and the interaction with other processes. Although the activities are shown with respect to time, however, in this case it is not according to any specific length of time but rather a sequential representation.

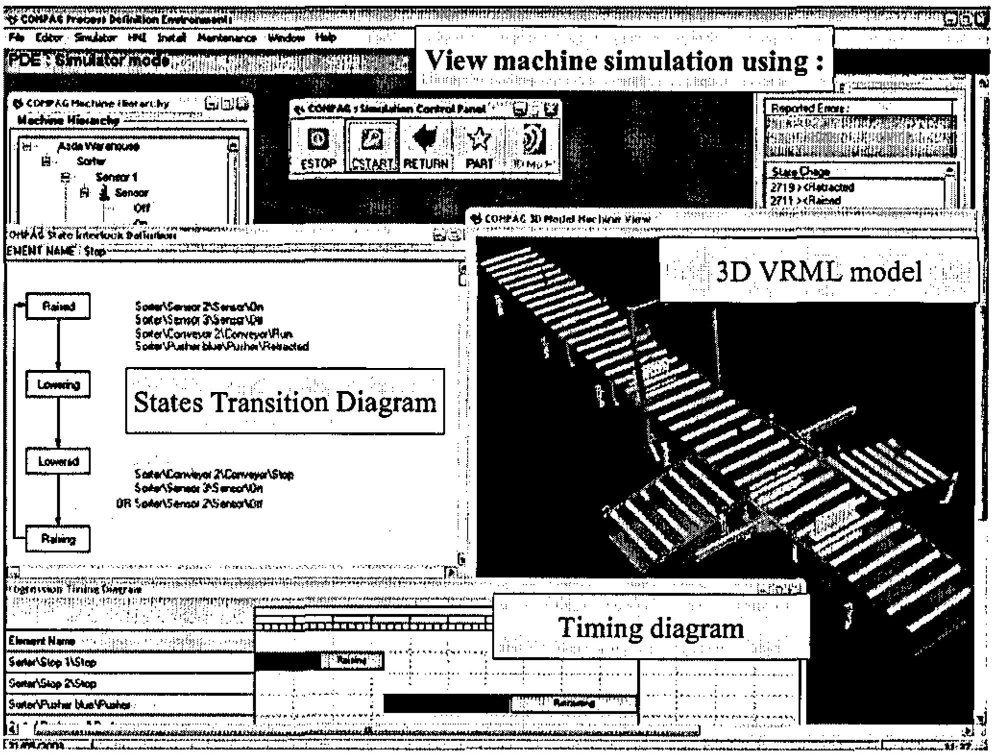


Figure A.1 Machine simulation in the PDE. The simulation could be viewed using the states transition diagram or the 3D VRML model. The simulation control panel in the top middle of the screen has the button for loading of parts and for starting the simulation.

3D VRML Model Creation

Figure A.5 shows the structure diagram for creating the 3D VRML Model, DP2. The activity diagram is shown in Figure A.6. The first process in the model creations is to analyse and understand the object to model (BP2.1). This process basically defines the architecture of the model. It involves defining the three aspects of model: i) geometry aspect (EA211) which predefined the type of motion, ii) behaviour aspect (EA212) which is the logical description of the states element, and iii) dynamic aspect (EA213), which describes the kinetic motion of the component. These three aspects are the basic elements within the architecture. As the basic elements, they are suitable for reused. The definition of the architecture elements (BP2.2) is to define these elements, which are: i) dynamic element (EA221), ii) state element (EA222), iii) condition element (EA223), iv) link point (EA224), v) view point (EA225) and vi) button for user interfaces (EA226). The above processes are essentially describing the building up of a library of reusable elements. If this library already exists, the user could proceed directly to the next process, which is 'Specific case description BP2.3'. This explains why the two processes 'BP2.1' and BP2.2 are shown as alternate flow in this diagram.

Under specific case description (BP2.3), the three main activities are i) knowing the mechanical parts (EA231), ii) knowing the moving parts (EA232), and iii) understanding the modelling requirement (BP231). In understanding the modelling requirement, it is essential to understand the level of details that the model is to present (EA2311), the information or the data required (EA2312) and the user's expectation from the modelling (EA2313). When these requirements have been understood, the basic shape of the model could be created (BP2.4). When creating the basic shape, if computer aided design (CAD) files are available (BP241), this will allow the exporting of the files to VRML (EA2411). If CAD files are not available (BP242), the basic shapes will have to be defined (EA2421) and then using any software package that can export them to VRML (EA2422).

Component configuration (BP2.5) is to populate the elements that defined the components. This includes populating the i) dynamic and the static element (BP251), ii) states element (BP253), iii) condition element, iv) link point and v) view point. Each of these processes are repeated as many times as required, depending on the number of elements involved. When populating the dynamic/static element (BP251), the position and the orientation of the components are to be determined (EA2511). This is followed by defining the kinematics parameter (BP252), which includes the i) axis of translation (EA2521), ii) axis of rotation (EA2522), and iii) default route (EA2523). To populate the states element (BP253), the following has to be defined: i) name of component (EA2531), ii) the integer position (EA2532) and iii) colour (EA2433). The process of populating the condition elements (BP254) defines: i) the name of the component (EA2541), ii) the route parameter (EA2542) and iii) the time relative to the speed and the position (EA2543).

When the component has been configured, they are ready for assembly (BP2.6). Firstly, it is the geometry assembly (BP261). The floor space for the component has to be defined (EA2611). The components to be assembled are selected (EA2612) and the link point are specified to join the components (EA2613). Following this process is the logical assembly (BP262); this involves the populating of the condition element (BP2621), in which the interlocks are defined (EA26211).

Create Component

The activities involved in the creation of a component under the Component Based Approach adopted by the CCG project is as shown in Figure A.7. This process (i.e. a domain process, DP3) will be of greater concern to the component suppliers than the other stakeholders (i.e. machine builder and end user).

The first process in creating a component is to decide the automation hardware (BP3.1). There are three activities under this process, that is to decide the type of the hardware (EA311), cycle of the component (EA312) and determine the behaviour of the hardware (EA313). The implementation of the control behaviour for the components is summarised under the BP 'implement control BP3.2'. The first sub-BP is to design the standard interface (BP3.21); under this process, there are several sub-BPs and EAs. The sub-BPs are implementing operation mode (BP3.22), designing electrical interface for inter-component connection (BP3.23), allocating storage space for different type of data (BP3.24), designing the network interface (BP3.25), developing the communication standard (Bp3.26), designing the proprietary code (BP3.27) and designing the internal electrical interface (BP3.28). These seven sub-BPs can be further decomposed to lower level activities, i.e. EAs.

The activities within designing of the standard interface are developing the states transition diagram (EA3211), specifying the distributed processor (EA3212), and implementing error routine (EA3213). This is followed by implementing the operation mode for the component (BP3.22), which are manual (EA3231), fully automatic (EA3232), semi automatic (EA3233) and debugging mode (EA3234). On designing the electrical interfaces to facilitate inter-component connection (BP3.23), the two activities are to set the cable standard (EA3231) and to design the wiring diagram (EA3232). Within the component, storage space has to be allocated for different type of data (BP3.24), which includes spaces for: i) interlocks (EA3241), ii) component parameters specific (EA3242) and iii) a look-up table (EA3243). When designing the network interface for the component (BP3.25), the communication standard has to be developed (BP3.26) so as to facilitate the downloading of the identity (EA3261).

The designing of the proprietary code (BP3.27) involves defining the application programmable interface (EA3271) and defining the look-up table (EA3272). This also involves designing the internal electrical interface (BP328). The two main activities under this process is to design the wiring diagram and to specify the input/output (I/O) for the component (EA3282).

Finally, the component is ready for final installation (BP3.3). During this last stage of the component development, the activities are to programme the state machine (EA331), testing (EA332) and debugging (EA333). The resources required for creating the components are shown in the activity diagram in Figure A.8.

Human machine interface (HMI)

The human machine interface (HMI) under the CCG project is based on web technologies and the layout and graphical display of the HMI are generated using templates. The machine builders are responsible for developing the HMI. Creating the HMI under the CB system is identified as a DP, 'Create HMI DP4', as shown in Figure A.9.

The first step in HMI creation is to understand the user's requirements (BP4.1). This will involve: i) identifying the user of the HMI (EA411), ii) identifying the usage of the HMI (EA412), iii) determining the information to display (EA413), and iv) determining the format of visualisation (EA414). The following process is to understand the operation procedure of the machine (BP4.2). This requires the i) understanding of the architecture of the system i.e. machine (EA4.21); ii) understanding the prioritisation of information (EA422), iii) understanding the manipulation requirement (EA423), and iv) determining the level of manipulation by different users (EA424).

A HMI pattern describes the functionality, user interaction and the capabilities of the HMI to perform common tasks. In defining the HMI pattern (BP4.3), the main activities are determining i) the HMI functions (EA431), ii) structuring of the information, iii) the strategy for manipulation and action (EA433), iv) diagnostic information display strategy (EA434) and v) navigation mechanisms (EA435). Once the HMI pattern has been defined, the designing of the HMI can be started (BP4.4). A template for all the HMI screens will be created (BP4.41) and the graphics that are to be used has to be created (EA441). A HMI template acts like a form; it establishes the basic layout of the HMI screens and is 'filled in' (or populated) individually according to the different information each screen is required to display. This means that this 'form' i.e. the templates are reusable. Thus, the items that are to be fixed on these screens have to be determined (BP4.42). This involves the determination of the layout of the screens (EA4421) and also the core view or the commonality between the screens. Next, the variant aspects on the HMI will be determined (BP4.43). This will include the determination of the i) user identification (EA4431), ii) screen colour (EA4432), and iii) the font type (EA4433).

The process of configuring the HMI is to populate the HMI with the components that make up the machine (B4.5). A configuration tool is required for this purpose, in the case of the CCG projects, this tool is provided in the PDE. The main activities during this process are server configuration (BP4.51) and specifying the display preferences (BP4.52). The machine server which holds the information of the components for the machine has to be specified (EA4511). On specifying the display preferences (BP452), the following items has to be specified: i) images that are to be used (EA4521), ii) the colour (EA4522), iii) the font to use (EA4523), and vi) the layout of the pages (BP4.53). This requires the specification of the number of components in each page (EA4531).

Figure A.10 is an activity diagram for HMI creation. It shows the different resources that are required for the creation of the HMI.

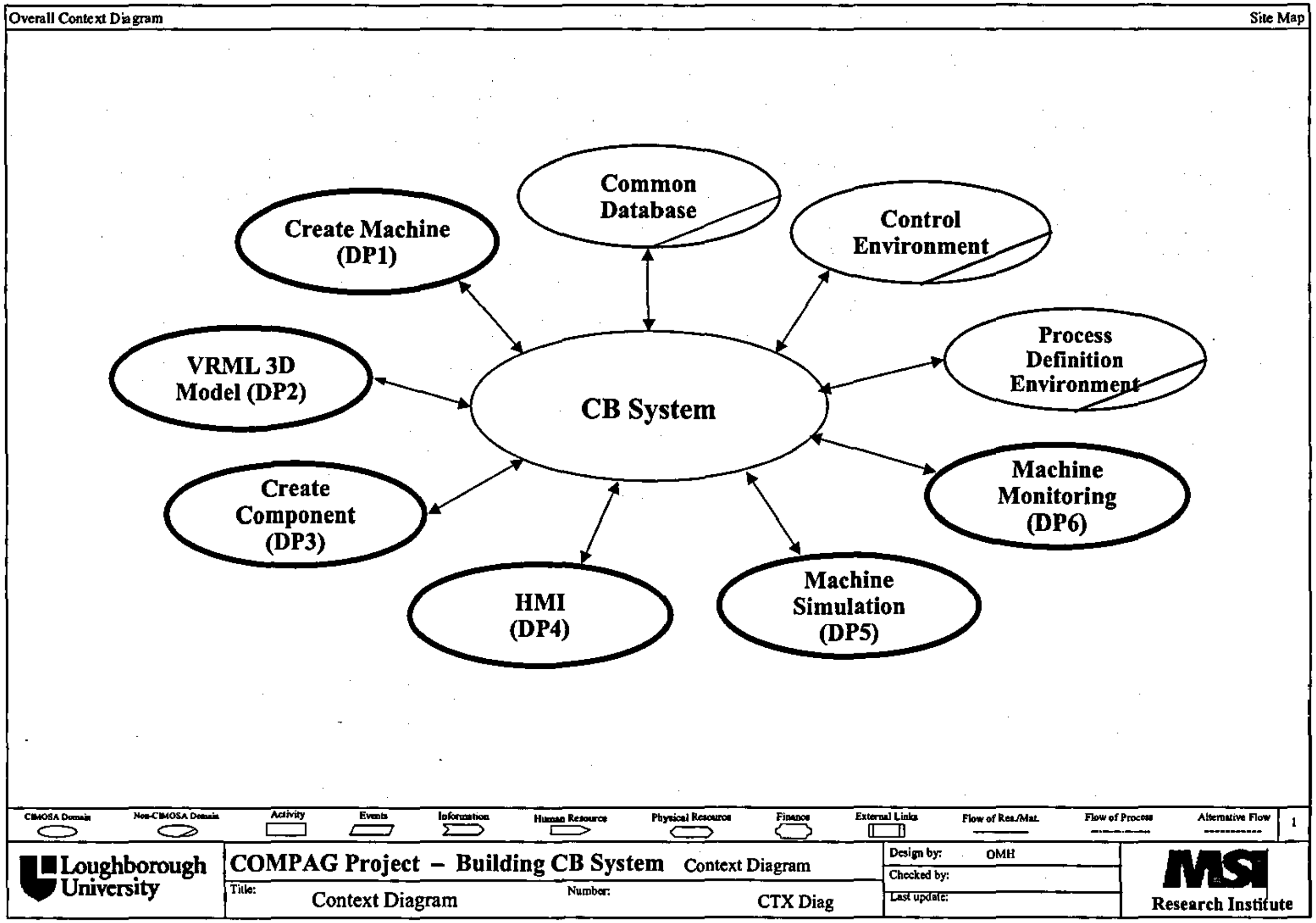


Figure A.2 Context Diagram for building CB System

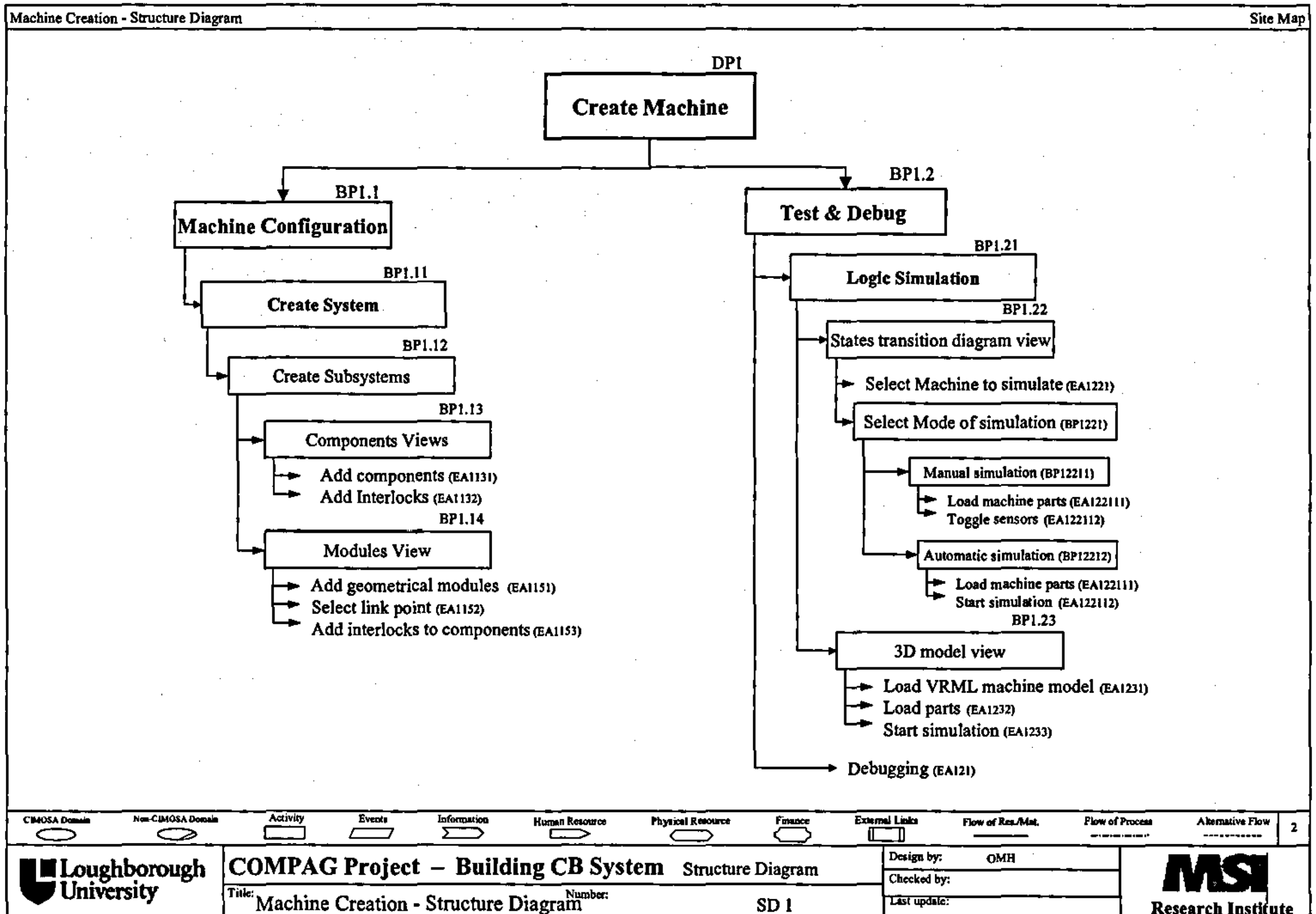


Figure A.3 Structure Diagram for Machine Creation

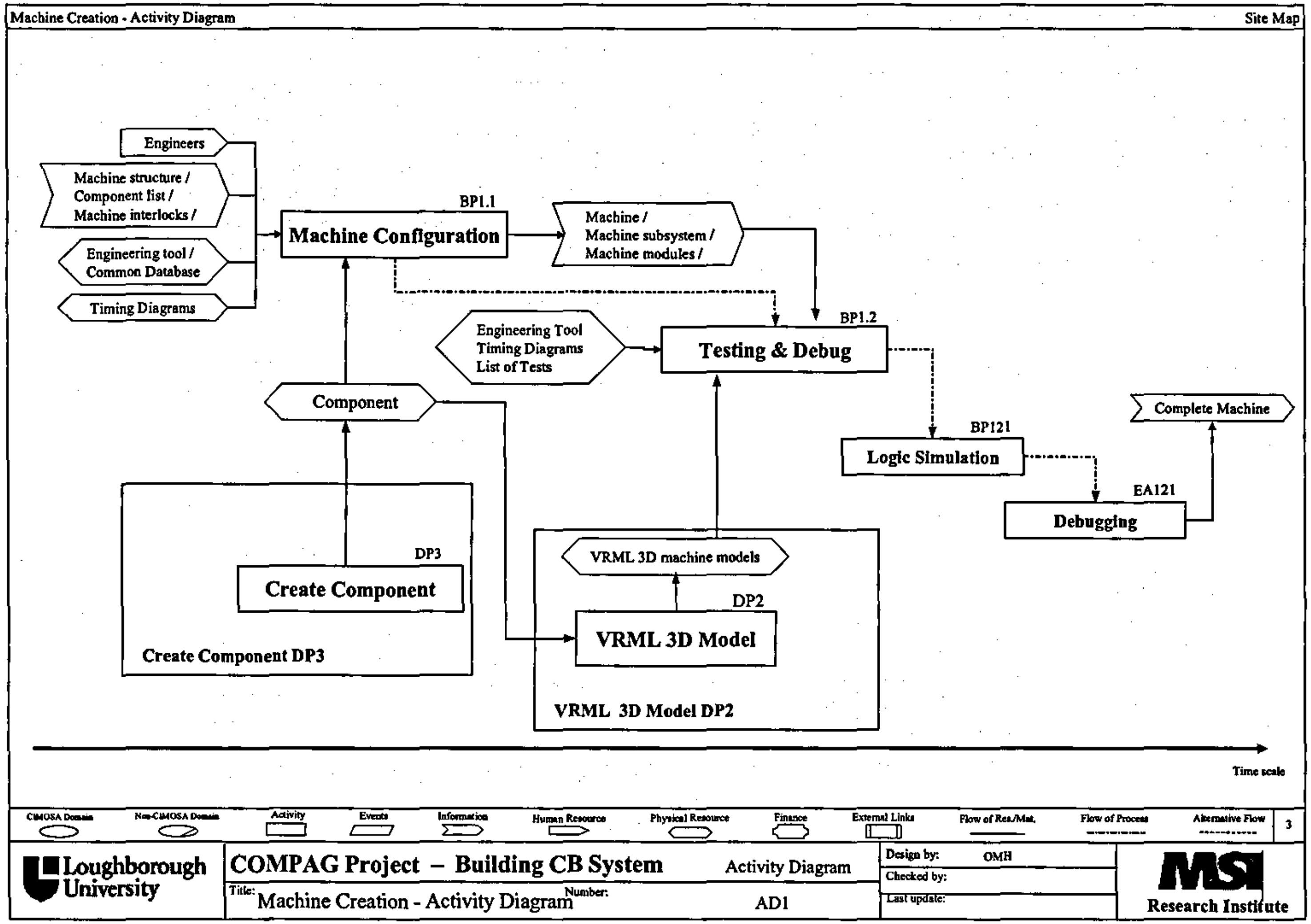


Figure A.4 Activity Diagram for Machine Creation

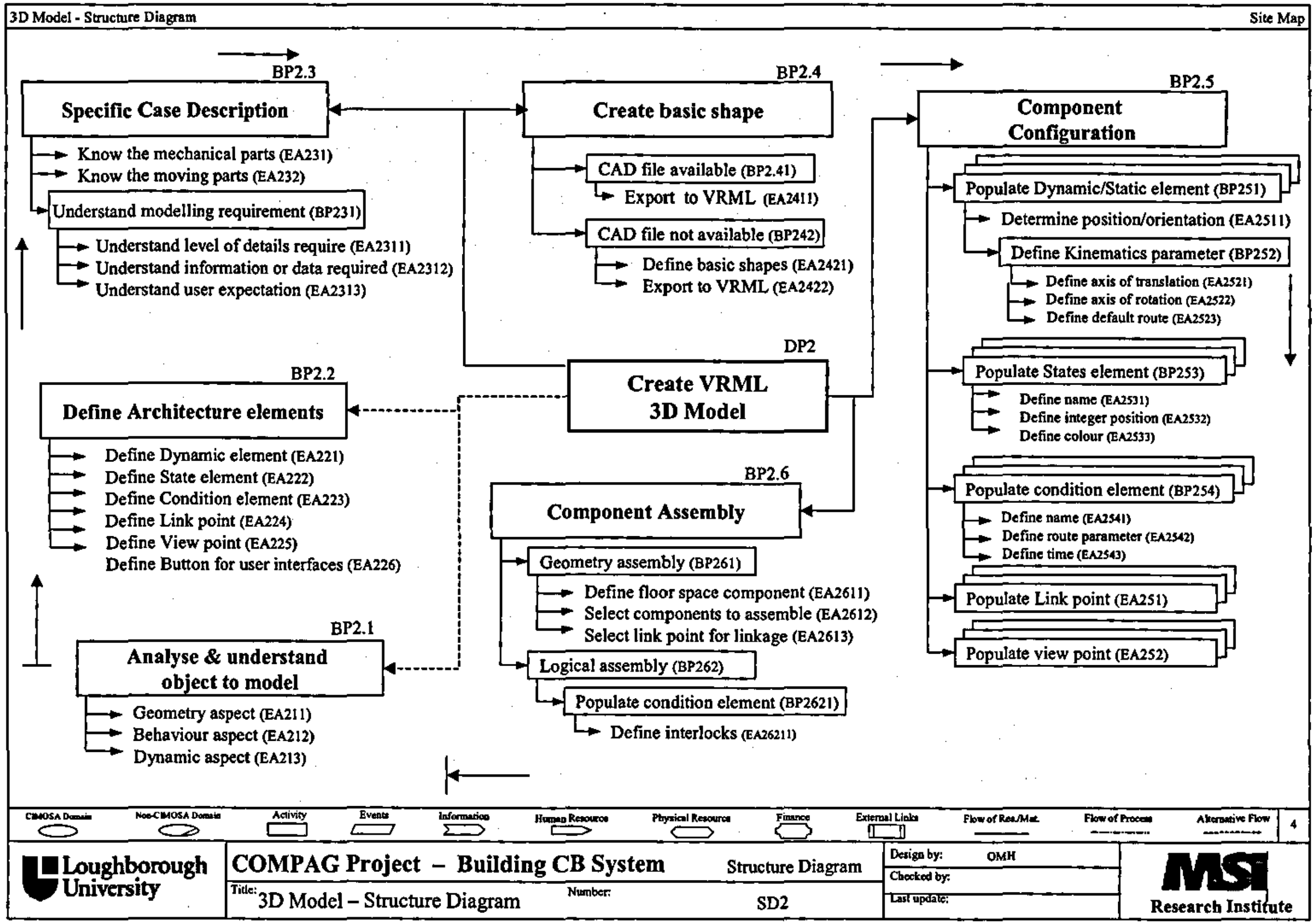


Figure A.5 Structure Diagram for 3D VRML Model Creation

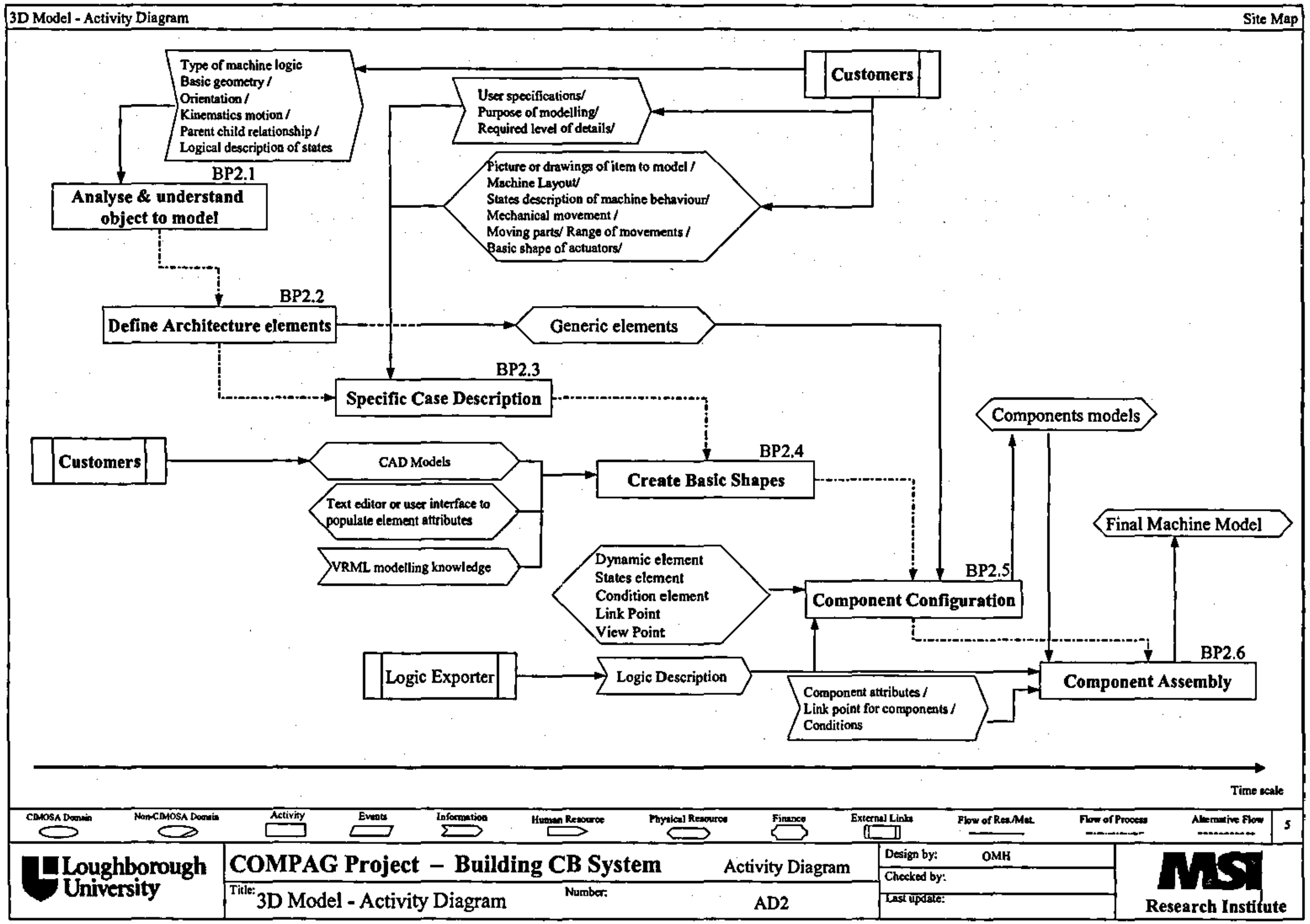


Figure A.6 Activity Diagram for 3D VRML Model Creation

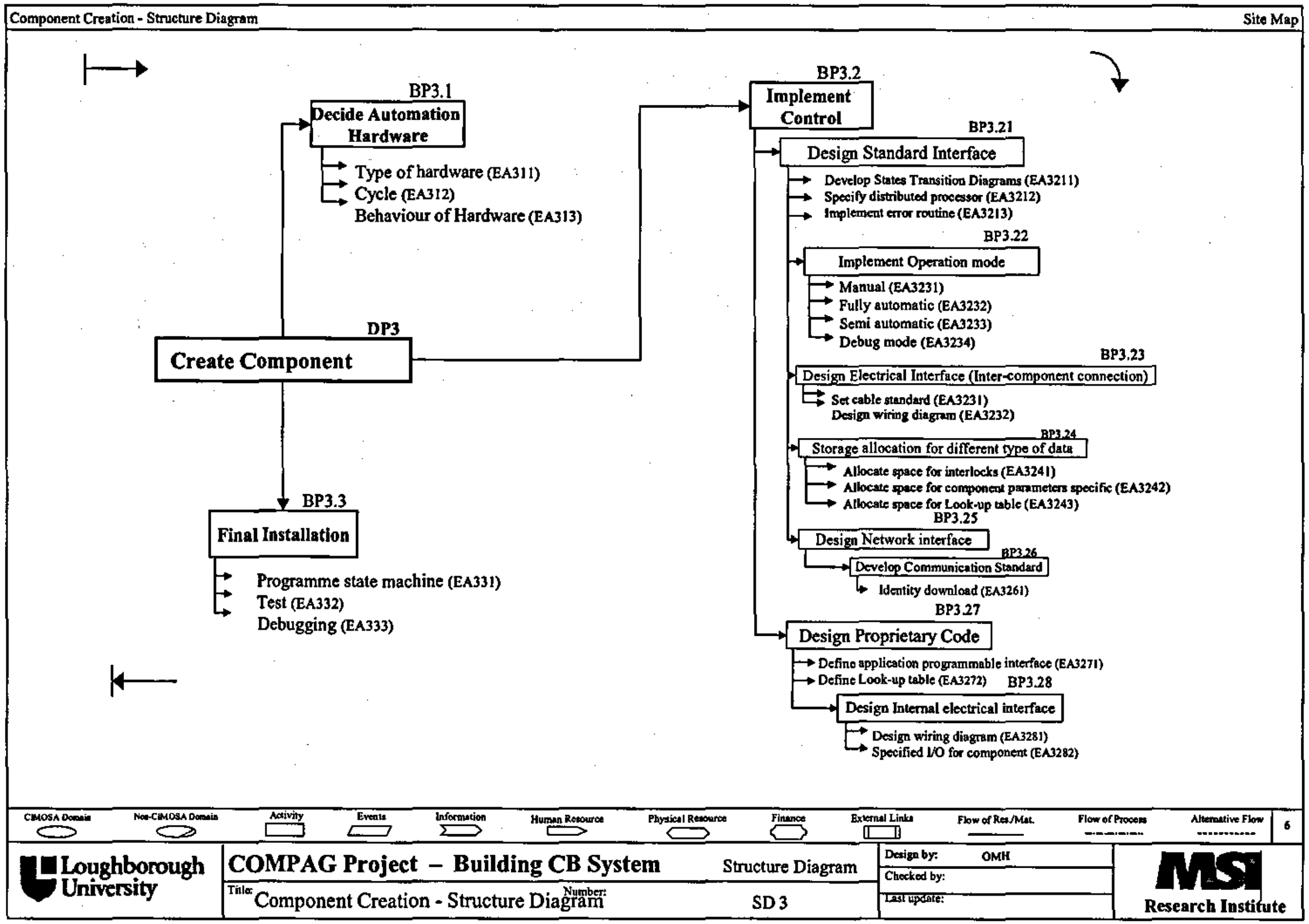


Figure A.7 Structure Diagram for Component Creation

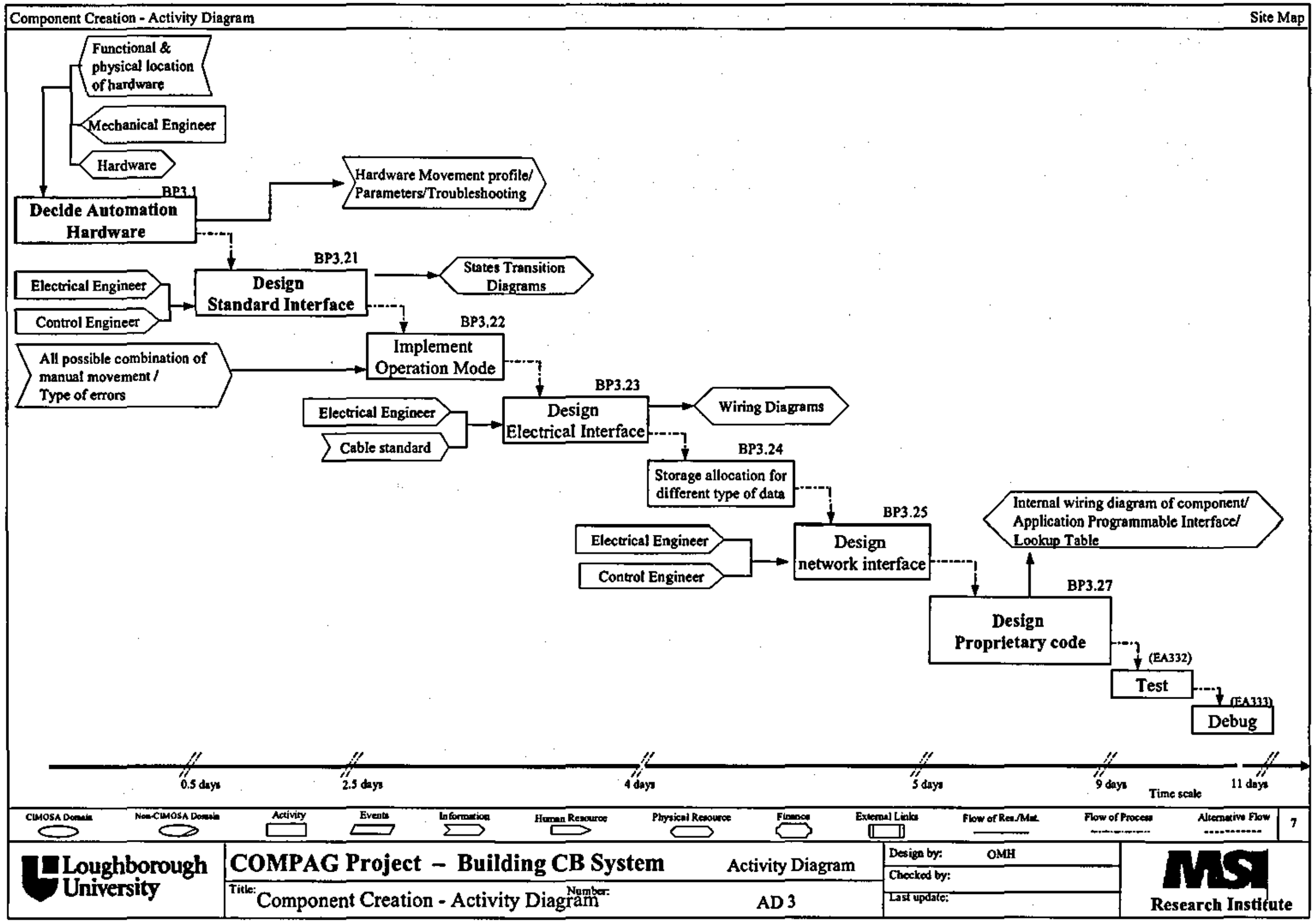


Figure A.8 Activity Diagram for Component Creation

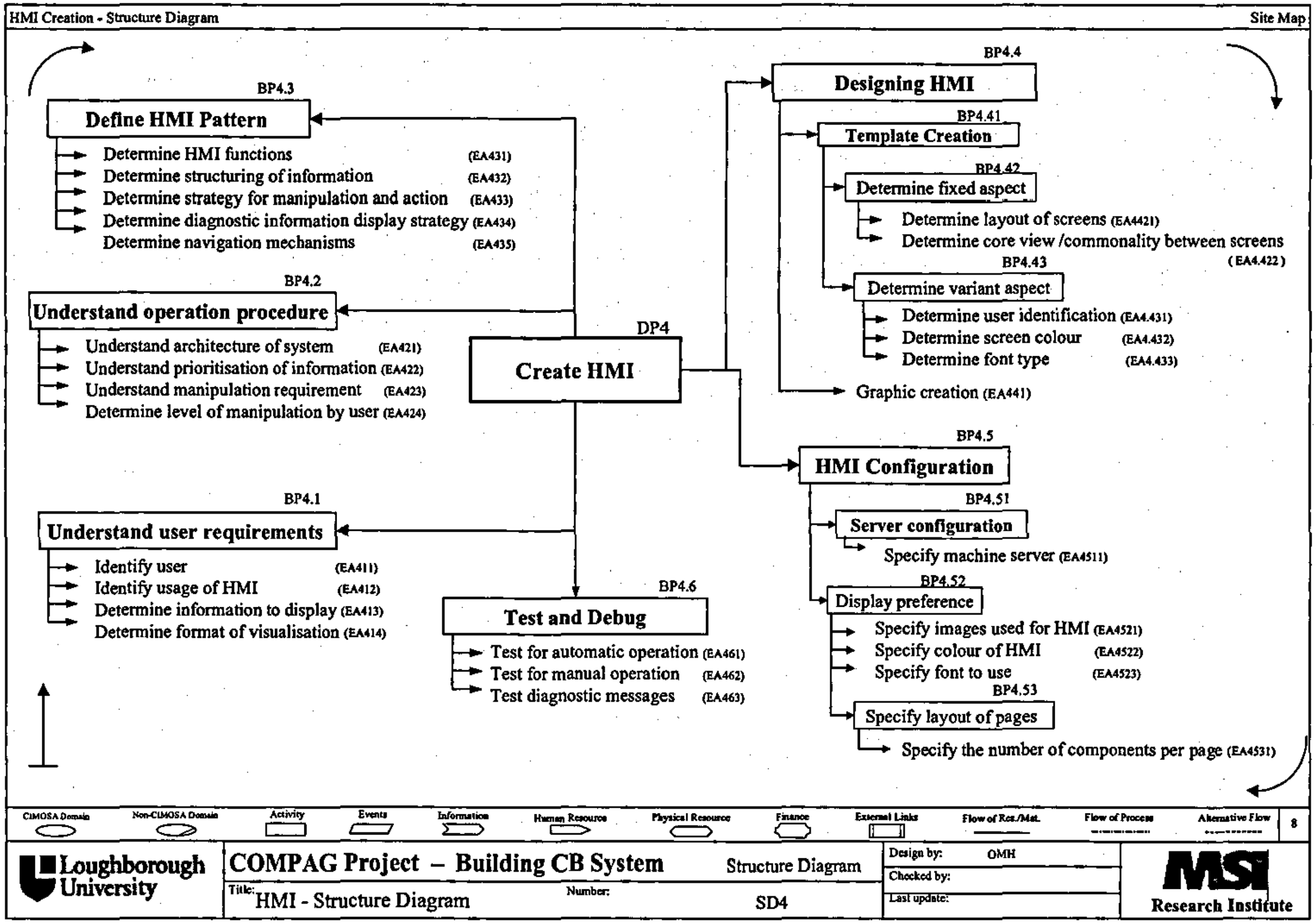


Figure A.9 Structure Diagram for HMI Creation

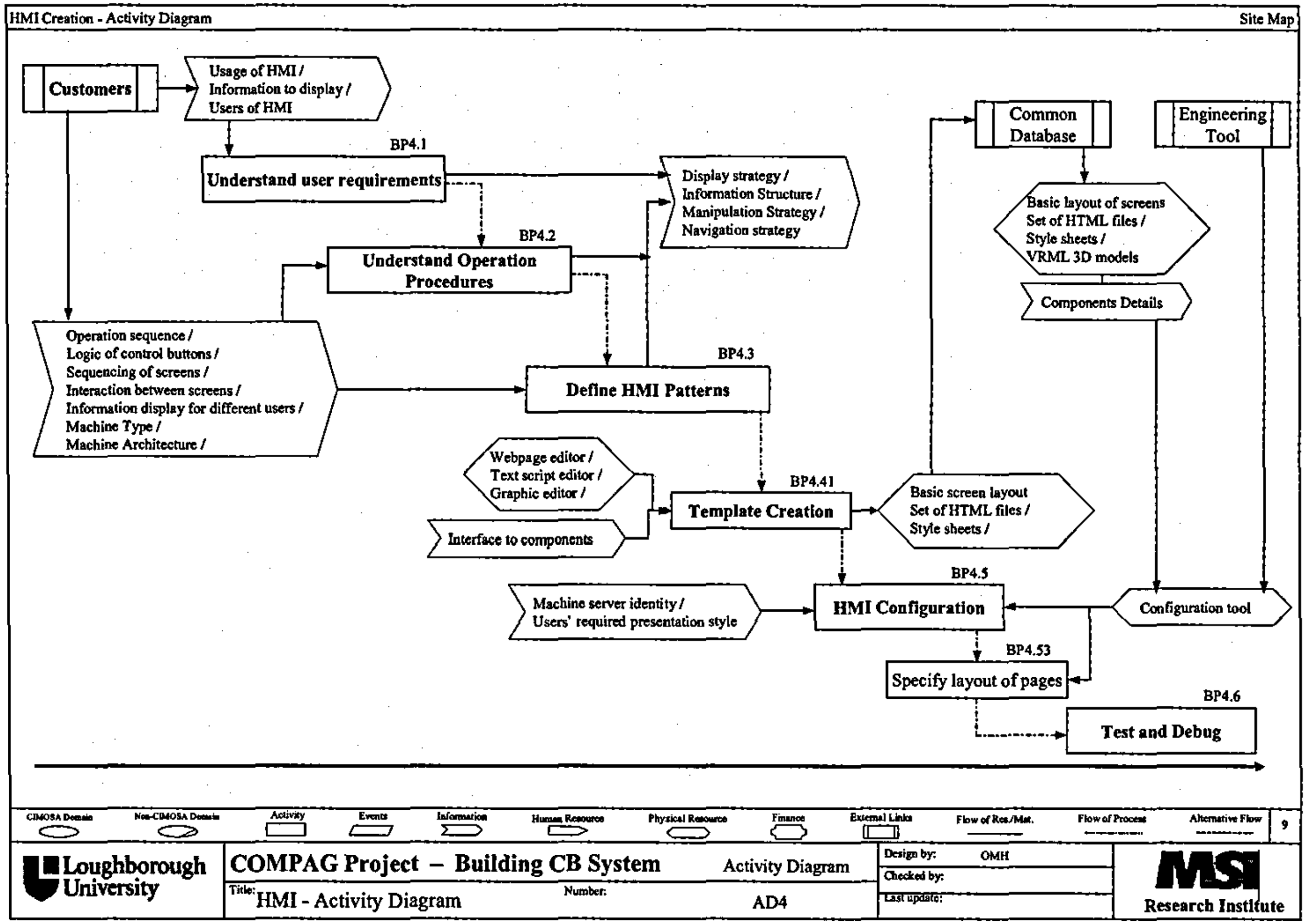


Figure A.10 Activity diagram for HMI Creation

Appendix B

Questionnaire and Survey Result

I) Questionnaire for Survey

Survey on Loughborough University's Component Based for Manufacturing Machinery

Purpose of Survey

This survey is carried out by the MSI Research Institute of Loughborough University, to gather feedback on the COMPAG project (i.e. COMPonent based Paradigm for AGile Automation). The project looks into the implementation of the Component Based Approach for the lifecycle engineering of machine automation.

Two test machines had been commissioned within the COMPAG project: i) an assembly machine at J.A. Krause, Germany, ii) a transfer line machine at Lamb Technicon, Britain. Project presentations and demonstrations were given at both Krause in June 2002, and recently at Lamb Technicon in February and April.

The agenda for the presentation was as followed:
i) concept of the Component Based Approach
ii) demonstration on the use of an engineering tool for machine configuration
iii) demonstration on 3D virtual simulation of machine
iv) demonstration on remote diagnostics
v) demonstration of the physical test machine at the shopfloor

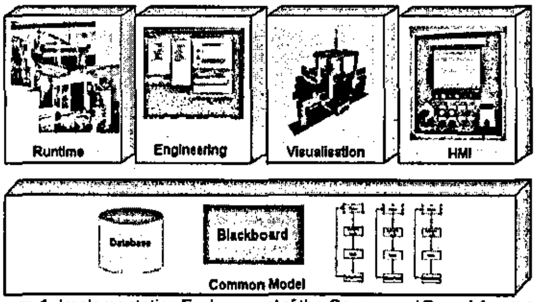


Figure 1. Implementation Environment of the Component Based Approach

Figure 1 illustrates the four basic environments within the Component Based Approach and the common model underpinning the environments. The runtime environment is the actual environment for executing the control logic to drive a machine; the engineering environment consists of the engineering tools for configuring the machines; the visualisation environment is where the virtual modelling and simulation of machines are viewed; the human machine interface (HMI) under the component based approach is based on Web technologies and enables access via a browser-based application interface for operation monitoring and diagnostics of an automation solution. The common model is a single common engineering environment for all stakeholders. It is supported by the use of a common database which stores all the data and information (such as project partners, project timeline, project updating, mechanical/electrical/control drawings) on the project. The data or information, defined and entered once, could be used many times throughout the lifecycle by all the stakeholders.

The Questionnaire

The questions in the questionnaire had been designed around the functionality developed by the project. The answers received will be collated and summarised. No individual details will be revealed in any reports on the study. The result will help the project group identify areas where i)improvement can be made, and ii) future work or development can be undertaken.

There are two parts to this questionnaire. The first part is on background information of the participant and the second part consists of questions relating to :
i) common model approach
ii) engineering tool
iii) visualisation tool
iv) human machine interface
v) comments on the COMPAG project

We appreciate your time and effort in completing this questionnaire. The questionnaire will take about 20 minutes to complete. Please look through the questions and choose an option which best describes your opinion.

On completing the questionnaire, please press the "Submit" button and the questionnaire will be sent via email.

Part I

Background Information

Name: __________

Email: __________

1) Which of the following group does your company belongs to:

☐ End-users

☐ Tier 1 supplier

☐ Tier 2 supplier

☐ Others (Please specify) __________

2) What appllcation domain (as your job) are you involved in (please tick as many as applicable):

- ☐ Aerospace
- ☐ Automotive
- ☐ Consumer products (non-food)
- ☐ Electronics/Semiconductor
- ☐ Fine chemical
- ☐ Food & beverage
- ☐ Machinery
- ☐ Metal fabrication
- ☐ Petrochemical
- ☐ Pharmaceuticals
- ☐ Retailing
- ☐ Transport
- ☐ Water & wastewater
- ☐ Wholesale or distribution
- ☐ Others

If you selected "Others", please specify:

3) What is your job responsibility?

4) What areas of automation are you involved in? (please tick as many as applicable)

- ☐ Software
- ☐ Hardware
- ☐ Engineering
- ☐ Design
- ☐ Finance
- ☐ Maintenance
- ☐ Others (please specify)

5) What engineering software tools do you currently use for your job? Please state why the tool is used.

6) What are the main problems (top 5) that you face during the process of machine design and implementation?

1
2
3
4
5

Part II

A) Common Model Approach

The common model is a single common engineering environment for all stakeholders. It is supported by the use of a common database which stores all the data and information (such as project partners, project timeline, project updating, mechanical/electrical/control drawings) on the project. The data or information defined and entered once, could be used many times throughout the lifecycle by all the stakeholders.

7) How would you rate the implementation of a common model approach?

○ Not useful ○ Less useful ○ Neutral ○ Useful ○ Very useful ○ Do not understand

8a) How useful would you rate the use of a common model in solving some of the problems faced during the process of machine building?

○ Not useful ○ Less useful ○ Neutral ○ Useful ○ Very useful ○ Do not understand

8b) Which problems would you consider as best solved by the common model?

B) Runtime Environment

The runtime environment refers to the control environment in the system. It is the actual environment for executing the control logic to drive a machine.

9) How important are the following factors (drivers) to decentralised (distributed) automation architecture ?

a) Increased Flexibility

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

b) Higher availability

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

c) Increased fault tolerance

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

d) Reduce complexity of control design and implementation

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

e) Simplified maintenance

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

f) Modularisation of machines

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

g) Scalable system

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

h) Reduced overall cost

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

i) Reduced overall time

○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

Additonal comments :

10) How important are the following factors to your organisation in the process of machine building (with respect to the definitions given) ?

a) Flexibility : the ability to accommodate changes

O Not important O Less Important O Neutral O Important O Very important

b) Reusability : the ability in which the machine's mechanical elements, actuators, sensors and control software can be reconfigured and reused

O Not important O Less important O Neutral O Important O Very important

c) Availability : the percentage of time a system is available when required

O Not important O Less important O Neutral O Important O Very important

d) Remote maintenance : the ability to provide maintenance service through the web

O Not important O Less important O Neutral O Important O Very important

e) Machine Modelling : using 3D virtual representation and animation of machine

O Not important O Less important O Neutral O Important O Very important

f) Design tools for visualisation : graphical visualisations that allows the engineering data to be represented in a form that is easy to interpret and interact with

O Not important O Less important O Neutral O Important O Very important

g) Design tools for simulation : graphical visualisations that allows the engineering data to be simulated

O Not important O Less important O Neutral O Important O Very important

h) Performance data of components : diagnostic information such as number of cycles, time to failure, error history etc. , embedded in the components

O Not important O Less important O Neutral O Important O Very important

i) Project management visability

O Not important O Less important O Neutral O Important O Very important

Additonal comments :

11) How would you rank your organisation in achieving the following factors (with respect to the definitions given)?

a) Flexibility : the ability to accommodate changes

O Bad O Below Average O Average O Good O Very Good O Not available

b) Reusability : the ability in which the machine's mechanical elements, actuators, sensors and control software can be reconfigured and reused

O Bad O Below Average O Average O Good O Very Good O Not available

c) Availability: the percentage of time a system is available when required

O Bad O Below Average O Average O Good O Very Good O Not available

d) Remote maintenance : the ability to provide maintenance service through the web

O Bad O Below Average O Average O Good O Very Good O Not available

e) Machine Modelling : using 3D virtual representation and animation of machine

O Bad O Below Average O Average O Good O Very Good O Not available

f) Design tools for visualisation : graphical visualisations that allows the engineering data to be represented in a form that is easy to interpret and interact with

O Bad O Below Average O Average O Good O Very Good O Not available

g) Design tools for simulation : graphical visualisations that allows the engineering data to be simulated

O Bad O Below Average O Average O Good O Very Good O Not available

h) Performance data of components : diagnostic information such as number of cycles, time to failure, error history etc. , embedded in the components

O Bad O Below Average O Average O Good O Very Good O Not available

i) Project management visability

O Bad O Below Average O Average O Good O Very Good O Not available

Additonal comments :

12) How would you rate the relevance of embedded control in achieving the following factors in machine building (with respect to the definitions given)?

a) Flexibility : the ability to accommodate changes

O Not important O Less important O Neutral O Important O Very important

b) Reusability : the ability in which the machine's mechanical elements, actuators, sensors and control software can be reconfigured and reused

O Not important O Less important O Neutral O Important O Very important

c) Availability: the percentage of time a system is available when required

O Not important O Less important O Neutral O Important O Very important

d) Remote maintenance : the ability to provide maintenance service through the web

O Not important O Less important O Neutral O Important O Very important

e)Machine Modelling : using 3D virtual representation and animation of machine

O Not important O Less important O Neutral O Important O Very important

f) Design tools for visualisation : graphical visualisations that allows the engineering data to be represented in a form that is easy to interpret and interact with

O Not important O Less important O Neutral O Important O Very important

g) Design tools for simulation : graphical visualisations that allows the engineering data to be simulated

O Not important O Less important O Neutral O Important O Very important

h) Performance data of components : diagnostic information such as number of cycles, time to failure, error history etc. , embedded in the components

- ○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

i) Project management visability

- ○ Not important ○ Less important ○ Neutral ○ Important ○ Very important

Additonal comments :

13) How would you rate the impact of embedded control on the following factors?

a) Cost

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

b) Time

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

c) Skill requirement on control design

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

d) Human resource requirement

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

e) Job responsibility of control engineers

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

f) Job responsibility of mechanical engineers

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

g) Job responsibility of electrical engineers

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

h) Job responsibility of commissioners

- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

Additonal comments :

14) What are the factors you would consider when deciding to take up embedded control and distributed system?

C) **Engineering Environment**

The engineeringenvironmentin the COMPAG project has been termed the Process Definition Environment PDE. It is an integrated engineering environment providingsupport at variousstages of machine building, from initial design of machine, control configuration, simulation and maintenance when the machine has gone into actual production.

15) How important are the following features in future automation programming environments (engineering tools)?

a) Ease of use

O Not important O Less important O Neutral O Important O Very important

b) Graphical programming

O Not important O Less important O Neutral O Important O Very important

c) Library of reusable Components

O Not important O Less important O Neutral O Important O Very important

d) Control platform independences

O Not important O Less important O Neutral O Important O Very important

e) Verification of control logic before commissioning

O Not important O Less important O Neutral O Important O Very important

f) Simulation of machine logic

O Not important O Less important O Neutral O Important O Very important

Additonal comments :

16) What other features of programming environments would you like to see?

17) How would you rate the engineering software tool that you are current using in your organisation in achieving the following features?

a) Ease of use

O Bad O Below average O Average O Good O Very good O Not available

b) Graphical programming

O Bad O Below average O Average O Good O Very good O Not available

c) Library of reusable Components

O Bad O Below average O Average O Good O Very good O Not available

d) Control platform independences

O Bad O Below average O Average O Good O Very good O Not available

e) Verification of control logic before commissioning

O Bad O Below average O Average O Good O Very good O Not available

f) Simulation of machine logic

O Bad O Below average O Average O Good O Very good O Not available

Additonal comments :

18) How would you rate the COMPAG's PDE in achieving the following factors?

a) Ease of use

O Bad O Below average O Average O Good O Very good O Not available

b) Graphical programming

O Bad O Below average O Average O Good O Very good O Not available

c) Library of reusable Components

O Bad O Below average O Average O Good O Very good O Not available

d) Control platform independences

O Bad O Below average O Average O Good O Very good O Not available

e) Verification of control logic before commissioning

O Bad O Below average O Average O Good O Very good O Not available

f) Simulation of machine logic

O Bad O Below average O Average O Good O Very good O Not available

Additonal comments :

19) Have you seen any other engineering software which has the similar functions as COMPAG's PDE? If yes, please go to Question (20), if no, please proceed to Question (21)

O Yes, please go to Question 20 O No, please proceed to Question 21

20) Please specify the name of the software and the company who produce the software

21) How would you rate the impact of a software product like the PDE on the following?

a) Cost of machine design
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

b) Time for machine design
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

c) Skill requirement for control design
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

d) Cost of Machine Building
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

e) Time of Machine Building
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

f) Human resource requirement
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

g) Cost of commissioning
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

h) Time of commissioning
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

i) Cost of machine maintenance
- ○ Large decrease ○ Decrease ○ No effect ○ Increase ○ Large increase

Additonal comments :

22a) What would be the best way to acquire a engineering software like the PDE for your organisation?

- ☐ Produce in-house
- ☐ Subcontract the work of producing/maintaining the software
- ☐ Providedby control system supplier (e.g Siemens, Schneider, Indramat etc)
- ☐ Buy as a commercial product from software house
- ☐ Others (Please specify)

22b) Please state your reasons behind your answer to Question 22(a)

D) Visualisation Environment
The visualisation environmentin this surveyrefers to the use of 3D virtualrepresentation and animation of a machine.

23) How important is a 3D virtual representation and animation of a machine for the following areas in the lifecycle of machine building?

a) Marketing/Sales
- O Not important O Less important O Neutral O Important O Very important

b) Simultaneous engineering
- O Not important O Less important O Neutral O Important O Very important

c) Mechanical Design
- O Not important O Less important O Neutral O Important O Very important

d) Control Design
- O Not important O Less important O Neutral O Important O Very important

e) Electrical Design
- O Not important O Less important O Neutral O Important O Very important

f) mechanical build of machine
- O Not important O Less important O Neutral O Important O Very important

g) Commissioning
- O Not important O Less important O Neutral O Important O Very important

h) Operation
- O Not important O Less important O Neutral O Important O Very important

i) Maintenance
- O Not important O Less important O Neutral O Important O Very important

j) Remote diagnostics
- O Not important O Less important O Neutral O Important O Very important

Additonal comments :

24) With regard to your organisation, what would be the impact of the COMPAG visualisation environment have on the following factors:

a) Cost of machine building
- O Large decrease O Decrease O No effect O Increase O Large increase

b) Time of machine building

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

c) Skill requirement

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

d) Human resource requirement

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

Additonal comments :

25) What would your concerns be in using 3D virtual representation and animation of a machine?

E) Human Machine Interface

The Human Machine Interface (HMI) under the COMPAG project is based on Web technologies and enables access via a browser-based application interface for operation monitoring and diagnostics of an automation solution. The layout and graphical display of the HMI is generated from templates.

26) How would you rate the implementation of web-based HMI ?

- [ ] Not useful
- [ ] Less useful
- [ ] Neutral
- [ ] Useful
- [ ] Very useful

27) Are there any concerns in the use of web-based HMI?

28) What would be the impact of a web-based HMI to the following factors:

a) Cost of machine building

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

b) Time of machine building

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

c) Skill requirement

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

d) Human resource requirement

- [ ] Large decrease
- [ ] Decrease
- [ ] No effect
- [ ] Increase
- [ ] Large increase

Additonal comments :

[ ]

29) How would you rate the automatic generation of HMI ?

O Not useful O Less useful O Neutral O Useful O Very useful

30) Are there any concerns in using automatically generated HMI?

[ ]

31) What would be the impact of a automatically generated HMI to the following factors:

a) Cost of machine building

O Large decrease O Decrease O No effect O Increase O Large increase

b) Time of machine building

O Large decrease O Decrease O No effect O Increase O Large increase

c) Skill requirement

O Large decrease O Decrease O No effect O Increase O Large increase

d) Human resource requirement

O Large decrease O Decrease O No effect O Increase O Large increase

Additonal comments :

[ ]

F) On COMPAG

32) How would you rate your interest in acquiring the following items provide in the COMPAG?

a) Common model approach

O Not interested O Less Interested O Neutral O Interested O Very interested

b) Distributed Control

O Not interested O Less Interested O Neutral O Interested O Very interested

c) Engineering tools

O Not interested O Less Interested O Neutral O Interested O Very interested

d) 3D visualisation tools

O Not interested O Less Interested O Neutral O Interested O Very interested

e) Web based HMI

O Not interested O Less Interested O Neutral O Interested O Very interested

f) Remote Monitoring

O Not interested O Less Interested O Neutral O Interested O Very interested

Additional comments:

33) How would you rate the results of COMPAG research group in the following area?

a) Research

O Bad O Below average O Average O Good O Very good

b) Practicality

O Bad O Below average O Average O Good O Very good

c) Potential scientific impact

O Bad O Below average O Average O Good O Very good

d) Potential benefits to society

O Bad O Below average O Average O Good O Very good

Additional comments :

34) Have you seen or know of any other project/work having the same concept/functionality as the COMPAG project? If yes, please specify the name of the project and the organisation.

O Yes

O No

If your answer is Yes, please specify the name of the project and the organisation

35) Given that the end-users (e.g automotive vendor) endorse the COMPAG approach, would your organisation adopt the approach?

O Yes

O No

Please states the reasons behind your answer.

End of Survey
Thank you for your time.

Submit

II)Survey Result

The first five questions are targeted at getting the background information of the respondents, which explains why they are not given in this document. This summary of the survey result will start with Question 6 of the questionnaire. Component supplier will be identified as CS, machine builder as MB and end user as EU

Q6 : Main Problems that are faced by the stakeholders during the process of machine design and implementation:

CS	MB	EU)
1) Complexity 2) Changing Technology	1) missing part information (drawing and try-out parts) 2) missing strategic information 3) missing purchase part information 4) missing information of time schedule and calculation 5) missing information existing own solutions	1) Getting machine builders to follow build and software structure standards 2) Getting plant buy-in to "not invented here" engineering 3) Process complexity driving machine control complexity 4) Hardware reliability for inter-nations /universal use 5) Integration of legacy controls with newer technology

Part II (A) : Common Model Approach

Q 7 : Usefulness of the implementation of a common model approach

CS	MB	EU
Very useful	Very useful	Do not understand

Q8 : Usefulness of a common model in solving the problems faced during the process of machine building.

CS	MB	EU
Very useful	Very useful	Do not understand

Q 8b: Problems best solved by using a common model

CS	MB	EU
Complexity	1) Missing strategic information 2) Missing purchase part information	1) Getting machine builders to follow build and software structure standards

	3) Missing information of time schedule & calculation 4) Missing information existing own solutions	2) Process complexity driving machine control complexity 3) Integration of legacy controls with newer technology

B) Runtime Environment

Q9 : Importance of following factors to decentralised automation architecture

	CS	MB	EU
a) Increased Flexibility	Neutral	Very important	Important
b) Higher availability	Less important	Important	Very important
c) Increased fault tolerance	Important	Important	Important
d) Reduce complexity of control design and implementation	Very important	Very important	Very important
e) Simplified maintenance	Important	Very important	Very important
f) Modularisation of machines	Less important	Very important	Very important
g) Scalable system	Less important	Important	Very important
h) reduced overall cost	Neutral	Very important	Important
i) reduced overall time	Important	Very important	Important

Q10: Importance of the following factors to the stakeholders organisations in the process of machine building

	CS	MB	EU
a) Flexibility	Neutral	Very important	Very important
b) Reusability	Important	Very important	Very important
c) Availability	Neutral	Very important	Very important
d) Remote maintenance	Important	Neutral	Neutral
e) Machine modelling	Neutral	Less important	Important
f) Design tools for visualisation	Important	Important	Important
g) Design tools for simulation	Important	Important	Important
h) Performance data of components	Important	Important	Neutral
i) Project management visibility	Neutral	Important	Important

Q11: Performance of the stakeholders' organisation in achieving the following factors:

	CS	MB	EU
a) Flexibility	Below average	Average	Good
b) Reusability	Average	Average	Average

c) Availability	Good	Very good	Good
d) Remote maintenance	Good	Not available	Not available
e) Machine modelling	Average	Not available	Not available
f) Design tools for visualisation	Average	Not available	Not available
g) Design tools for simulation	Below average	Not available	Not available
h) Performance data of components	Below average	Very good	Not available
i) Project management visibility	Below average	Good	Good

MB's comment : Not available means have not been used.

Q12 : The relevance of embedded control in achieving the following factors for machine building :

	CS	**MB**	**EU**
a) Flexibility	Important	Very important	Important
b) Reusability	Neutral	Very important	Important
c) Availability	Important	Very important	Very important
d) Remote maintenance	Important	Neutral	Neutral
e) Machine modelling	Neutral	Neutral	Important
f) Design tools for visualisation	Important	Very important	Important
g) Design tools for simulation	Important	Important	Important
h) Performance data of components	Neutral	Important	Neutral
i) Project management visibility	Important	Important	important

Q13 : Rate the impact of embedded control on the following factors :

	CS	**MB**	**EU**
a) Cost	No effect	Decrease	Decrease
b) Time	Decrease	Large decrease	Decrease
c) Skill requirement on control design	No effect	Decrease	Increase
d) Human resource requirement	No effect	No effect	Increase
e) Job responsibility of control engineers	Increase	No effect	No effect
f) Job responsibility of mechanical engineers	No effect	No effect	No effect
g) Job responsibility of electrical engineers	Increase	No effect	No effect
h) Job responsibility of commissioners	No effect	No effect	No effect

MB's comments : for (e) to (h), 'no effect' means the responsibility remains the same (only the tasks are different).

EU's comments : Increase in resources would be needed initially but after full integration and one complete project using the tools/architectures, resource needs may reduce.

Q14 : Factors that would be considered when deciding to take up embedded control and distributed system.

CS	i) Commercial availability, ii) requirement by automotive manufacturers
MB	i) cost reduction, ii) proven components from a major vendor, iii) one look and feel style for the whole project.
EU	i) Life time cost, ii) ease of use, iii) reliability, iv) reconfigurability

C) Engineering Environment

Q15 : Importance of the following features in future automation programming environments (engineering tools)

	CS	MB	EU
a) Ease of use	Important	Very important	Very important
b) Graphical programming	Important	Important	Very important
c) Library of reusable components	Important	Very important	Very important
d) Control platform independences	Neutral	Important	Neutral
e) Verification of control logic before commissioning	Neutral	Important	Important
f) Simulation of machine logic	Neutral	Neutral	Important

MB's comment : proven components have to be available easily

Q16 : Other desirable features of programming environments

CS	No comments
MB	i) cost estimation ii) electrical drawings iii) integrated customer documentation
EU	No comments

Q17 : Rate the engineering software tool that the organisation is currently using on the following factors :

	CS	MB	EU
a) Ease of use	Average	Below average	Not involved
b) Graphical programming	Below average	Below average	Not involved
c) Library of reusable components	Average	Bad	Not involved
d) Control platform independences	Good	Bad	Not involved
e) Verification of control logic before commissioning	Below average	Not available	Not involved
f) Simulation of machine logic	Below average	Not available	Not involved

Note : The EU respondent has not been involved in using the engineering software.

Q18 : Rate COMPAG's PDE in achieving the following factors :

	CS	MB	EU
a) Ease of use	Good	Good	Not available
b) Graphical programming	Good	Good	Not available
c) Library of reusable components	Average	Average	Not available
d) Control platform independences	Below average	Not available	Not available
e) Verification of control logic before commissioning	Good	Good	Not available
f) Simulation of machine logic	Good	Good	Not available

Q19 : Have you seen any other engineering software which has the similar functions as COMPAG's PDE ?

CS	No
MB	Yes
EU	No

Q 20 : Other engineering software which has the similar functions as COMPAG's PDE

MB : Siemens Xmat and Siemens em-PLC

Q21 : Rate the impact of a software product like PDE on the following :

	CS	MB	EU
a) Cost of machine design	Decrease	Large decrease	Decrease

b) Time of machine design	Decrease	Large decrease	Decrease
c) Skill requirement for control design	No effect	No effect	No effect
d) Cost of machine building	No effect	Decrease	Decrease
e) Time of machine building	No effect	Decrease	Decrease
f) Human resource requirement	Decrease	No effect	Decrease
g) Cost of commissioning	Decrease	Large decrease	Decrease
h) Time of commissioning	Decrease	Large decrease	Decrease
i) Cost of machine maintenance	No effect	Decrease	Decrease

Q22(a): What would be the best way to acquire an engineering software like the PDE:

CS	Buy as commercial product from software house Others : Need to be supplied by major control supplier
MB	Others : A combination of all given choices would fit best
EU	Buy as commercial product from software house Others : Partnerships with supplier

Q22(b):

CS	Need full support, on site, around the world
MB	All external designed tools have to be integrated into the existing in-house solution. Tools for design and programming have to be accepted by and must be available for end-customers.
EU	Ideally a product could be purchased that is fully functional and will run on current PCS and servers. Alternative and not preferred as much, a "special" system could be developed as a partnership.

D) Visualisation Environment

Q23 : Importance of a 3D virtual representation and animation of a machine for the following areas in the lifecycle of machine building:

	CS	MB	EU
a) Marketing/Sales	Less important	Less important	Important
b) Simultaneous engineering	Neutral	Important	Important
c) Mechanical design	Important	Important	Important
d) Control design	Important	Important	Important
e) Electrical design	Less important	Not important	Neutral
f) Mechanical build of machine	Less important	Not important	Important
g) Commissioning	Neutral	Not important	Important
h) Operation	Neutral	Not important	Neutral
i) Maintenance	Important	Not important	Neutral

j) Remote diagnostics	Important	Important	Neutral

Q24 : Impact of COMPAG visualisation environment to the organisation on the following factors :

	CS	MB	EU
a) Cost of machine building	Decrease	Not effect	Decrease
b) Time of machine building	No effect	No effect	Decrease
c) Skill requirement	No effect	No effect	Decrease
d) Human resource requirement	No effect	No effect	Decrease

MB's comment : No effect : Tools are not integrated into in-house solutions and do not support used control systems

Q25 : Concerns in using 3D virtual representation and animation of a machine

CS	No comment
MB	Nice to have, if no additional cost or work required
EU	Time to develop/ resources needed

E) Human Machine Interface

Q26: Rate the implementation of a web-based HMI

CS	MB	EU
Useful	Neutral	Very useful

Q27 : Concerns on the use of web-based HMI

CS	No comment
MB	Nice to have, if web technology is available in the project (currently not)
EU	Reliability and speed of communications networks

Q28: Impact of web-based HMI on the following factors:

	CS	MB	EU
a) Cost of machine building	No effect	No effect	Decrease
b) Time of machine building	Decrease	No effect	No effect
c) Skill requirement	No effect	No effect	No effect
d) Human resource requirement	No effect	No effect	No effect

MB's comment : effct in control design and commissioning

Q29: Rate the automatic generation of HMI

CS	MB	EU
Useful	Useful	Useful

Q30: Concerns about the use of automatically generated HMI

CS	No comment
MB	How could customer requirements be taken into consideration
EU	No comment

Q31: Impact of a automatically generated HMI on the following factors:

	CS	MB	EU
a) Cost of machine building	No effect	No effect	Decrease
b) Time of machine building	Decrease	No effect	Decrease
c) Skill requirement	No effect	No effect	No effect
d) Human resource requirement	No effect	No effect	No effect

MB's comment : effects in control design and commissioning

F) On COMPAG

Q32: Rate your interest in acquiring the following items provided in COMPAG :

	CS	MB	EU
a) Common model approach	Less interested	Very interested	Interested
b) Distributed control	Less interested	Very interested	Very interested
c) Engineering tools	Not interested	Very interested	Interested
d) 3D visualisation tools	Not interested	Interested	Interested
e) Web based HMI	Not interested	Very interested	Interested
f) Remote monitoring	Not interested	Interested	Neutral

Q33: Rate the results of COMPAG research group in the following area:

	CS	MB	EU
a) Research	Good	Good	Good
b) Practicality	Good	Very good	Good
c) Potential scientific impact	Below average	Good	Good
d) Potential benefits to society	Average	Good	Good

Q34: Have you seen or know of any other project/work having same concept/functionality as the COMPAG project?

CS	NO
MB	Yes i) University Stuttgart ii) www.foederal.org iii) www.infoteam.de
EU	Yes

Q35: Given that the end users endorse the COMPAG approach, would your organisation adopt the approach

CS	Yes I assume that it would then be available commercially
MB	Yes Our company participated in the project from the beginning and supports LU as much as possible. Parallel, we set up an own internal project, to do our internal work to be ready for an integration of the whole engineering process.
EU	Yes

Appendix C

Documentation on Lamb Technicon

I) Understanding Process in Lamb Technicon

The typical period for the development of one transfer line is about 48-50 weeks but due to market competition and pressure, customers have requested Lamb Technicon (refer to as Lamb) to reduce the development time to 42 weeks. Lamb conduct simultaneous engineering with their customers; an effective SE will take about six months but in most cases, due to time pressure, they can only have two months for SE. When Lamb is invited to SE by Ford, it is based on the understanding that the contract will be awarded to them. This shows the close working relationship between Lamb and Ford.

At Lamb Technicon UK, they sub-contracted most of the work for mechanical design and building of the mechanical parts. The mechanical department at this site has limited capacity; they are able to handle tooling design for the machine but not for the design of a whole transfer line. They work closely with their sub-contractors; the sub-contractors are involved in a project from the start of SE, providing the conceptual design of the machine.

When Ford initiate a new engine project, Lamb will be invited to tender to design and build the production machinery. Lamb is also invited for the simultaneous engineering of the new engine block. These meetings are usually attended by the engineers, mostly mechanical engineers, from the Marketing and Sales department. During these meetings, Ford will provide Lamb with the new engine parts drawings. Based on these parts drawings, Lamb will design the engine production machine and determine the contract proposal details. In most cases, the new machine is designed based on designs from previous projects, with necessary modification tailored to the requirement for the new engine cylinder block. Lamb will sub-contract the mechanical design work and the sub-contractor will provide the conceptual design of the machine and this is then presented to Ford.

When the contract proposal has been approved and the sale order has been issued, the Marketing and Sales engineers will produce a document, known as the "Green Book" for internal circulation. The Green Book contains all the details about the deal, the machine specifications, the financial details, project schedules etc. The issue of the sale order will also market the start of the project. A meeting is then held within Lamb to announce the new project officially and the Marketing and Sales department will hand the project to the Project Management department. This phase is known as the Pre-engineering phase, in which the resources are allocated for the project, including human, financial, physical, logistical resources. Most of the mechanical design and building work are sub-contracted; the engineers from the mechanical department work closely with the sub-contractors, at the same time monitoring the development work. As the project enters the engineering phase, the control department also starts to design the control, electrical and hydraulic system. Once the design of the machine is completed, the manufacturing of the parts begins. The assembly of the machine begins when the parts are manufactured. At this stage, Lamb's technician will assemble the smaller parts and put in the associated electrical wiring; the work for assembling the base of the machine and the major electrical wiring work are sub-contracted involving sub-contract labourers.

The commissioning of the machine comes in after the machine has been assembled. According to Lamb, commissioning is the only time to verify the machine design after months of effort. Commissioning of the machine includes the downloading of the control software and testing, which is known as "Shakedown". This usually involves one commissioner and one control engineer. After commissioning, the machine would have achieved "autocycle", i.e. the machine is able to run properly without any parts, also known as the "dry run". This is followed by testing and adjustment of the machine. Parts are supplied by Ford for the testing. The purpose of this process is to make sure that the work quality meets the production requirement of Ford. At this phase, Lamb also have to check that the production speed of the machine meets the requirement, i.e. run capability. Ford will send its engineers to check that the requirements are met before approving the project, i.e. acceptance. Once Ford has accepted the test results, Lamb will be able to tear down the machine and prepared for shipment.

To install the machine at Ford's plant, Lamb will employ sub-contract labourers for the installation work. When the machine has been installed and tested, it is ready for production, i.e. Job 1. For a transfer machine, Lamb will usually deploy about twenty engineers. This is reduced to two engineers, (one mechanical and one control engineer) once the machine is running smoothly.

In terms of handling customer changes, Lamb's has a set of change management policy. The changes are categorised as "Post Order Changes" and "Non Post Order Changes". "Post order changes" are the changes requested by the customer and "Non Post Order Changes" are changes initiated by the machine builder and are not chargeable to the customer. When a change is requested by the customer, Lamb will assess what will be the impact of the change to the whole machine and make a cost estimation. If the customer agrees with the quotation for the change, the change will proceed. In most cases, Non-POC are minor changes and would not have much impact on the overall project. POC, on the other hand, will have an impact on the project development, resulting in changes such as design changes, changes in budget or changes to project schedule etc.

II) CIMOSA based process models for Lamb

The CIMOSA based process models for Lamb are shown in Figure C.1 to Figure C.5.

III) IThink™ Model for Lamb

Please refer to the CD-ROM attached to the thesis.

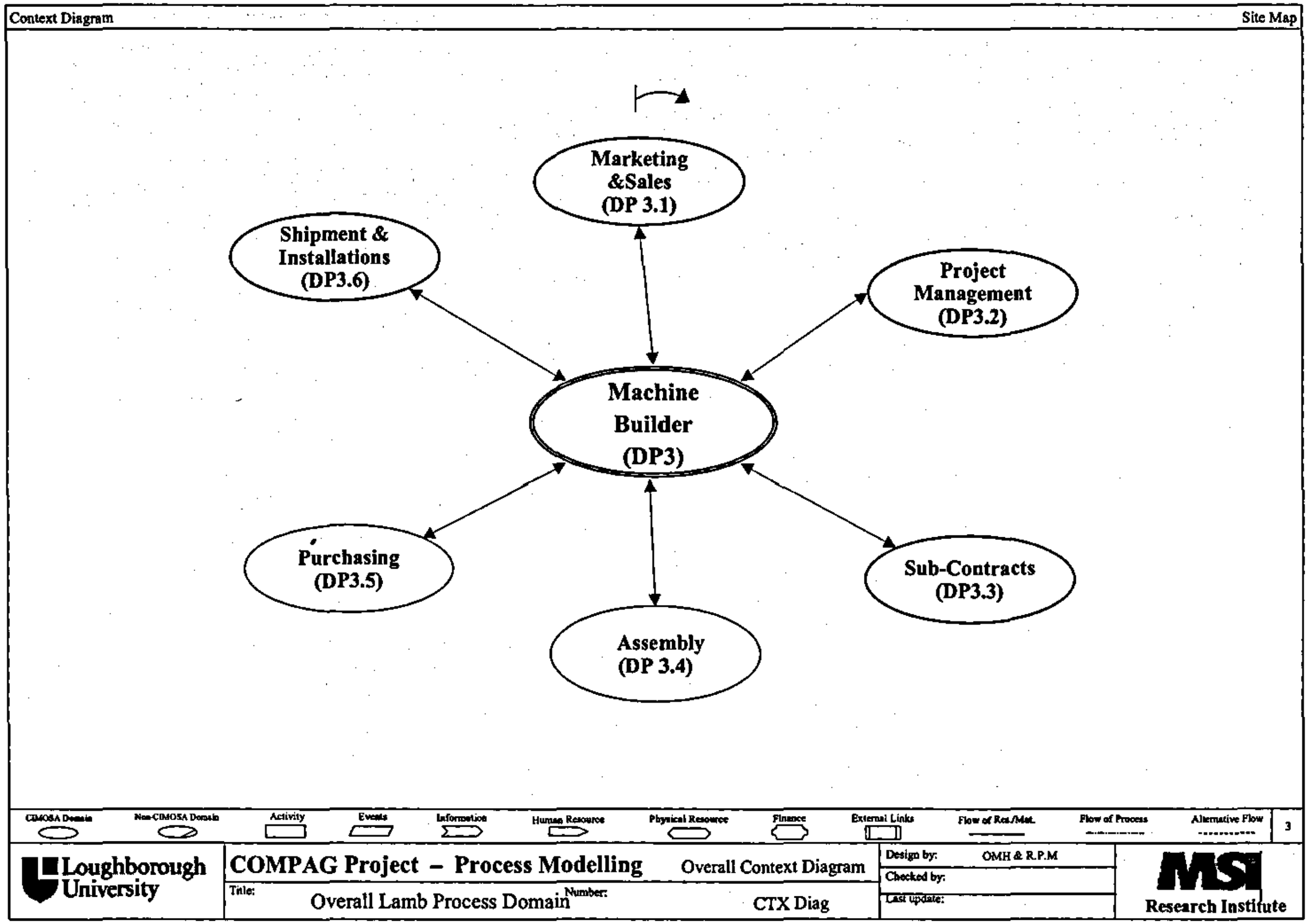


Figure C.1Context Diagram for Lamb

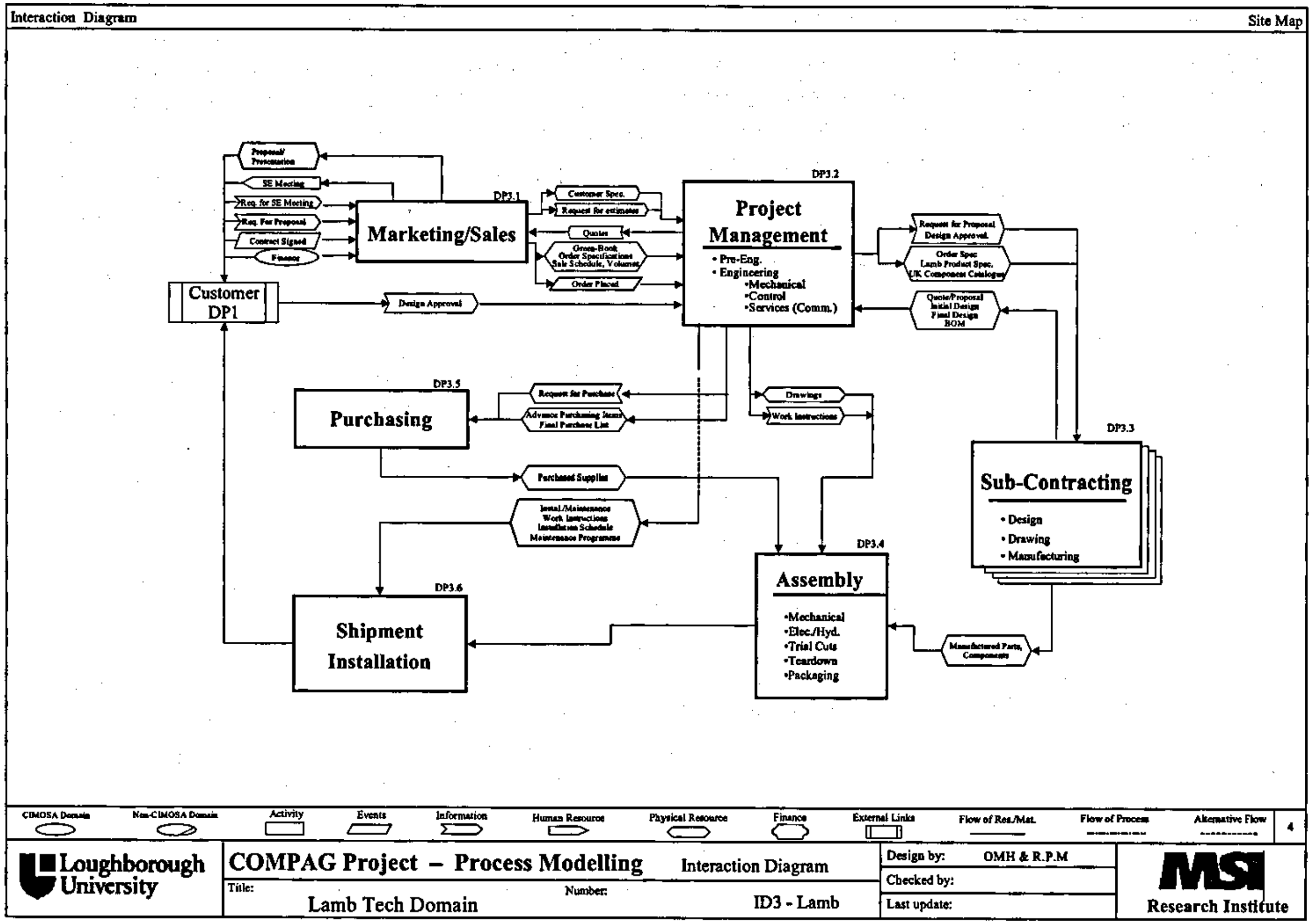


Figure C.2 Interaction Diagram for Lamb

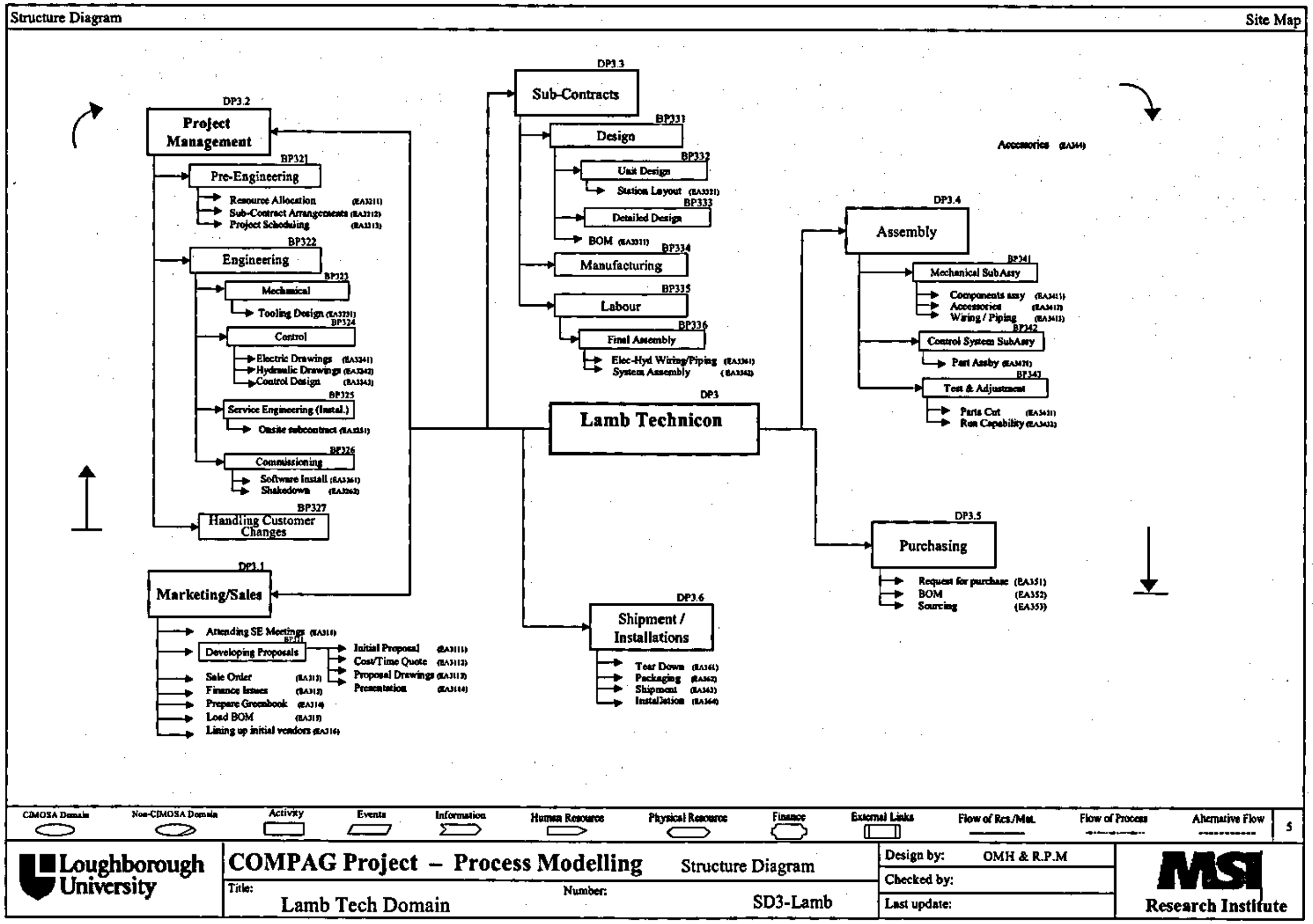


Figure C.3 Structure Diagram for Lamb

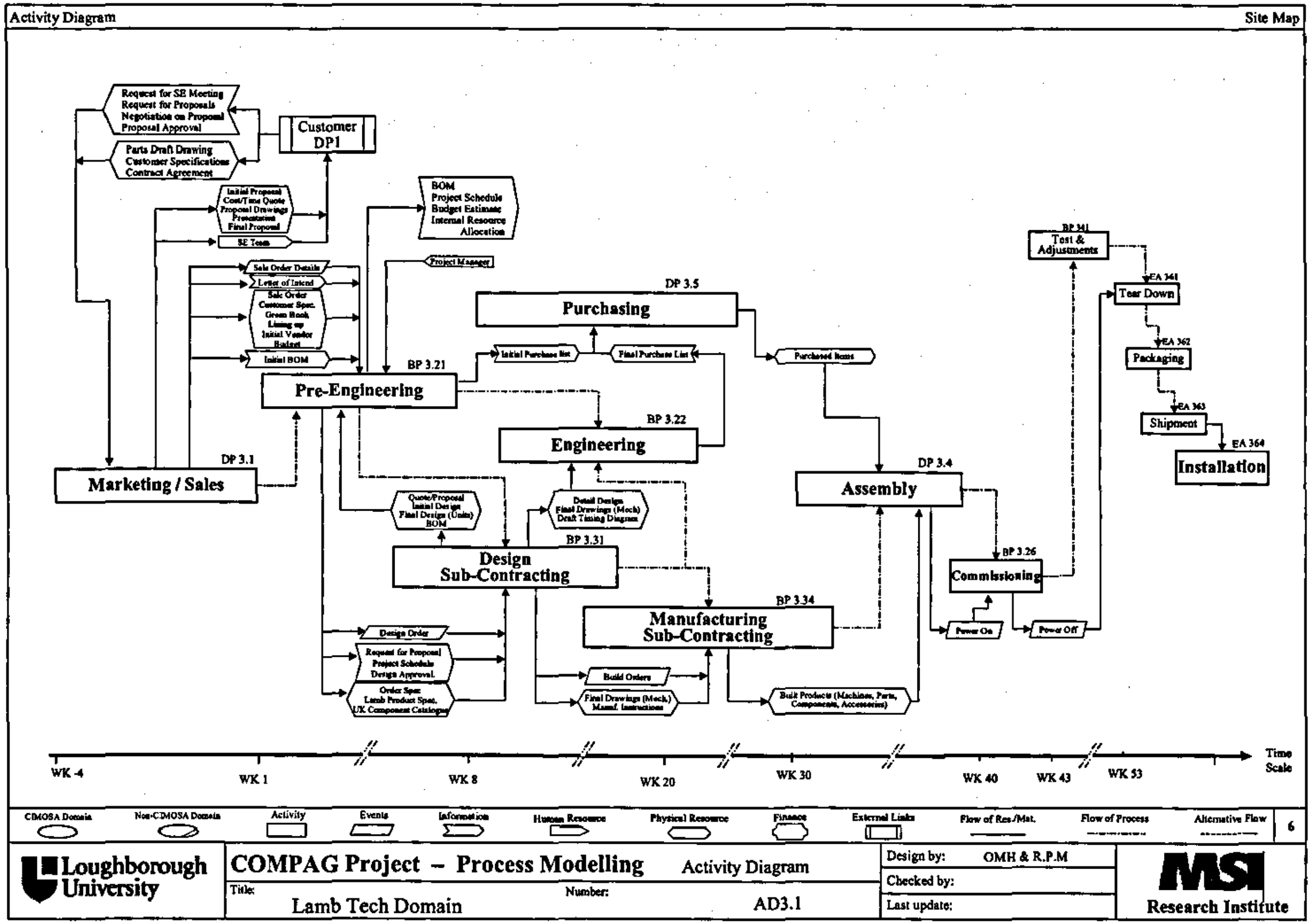


Figure C.4 Activity Diagram (I) for Lamb

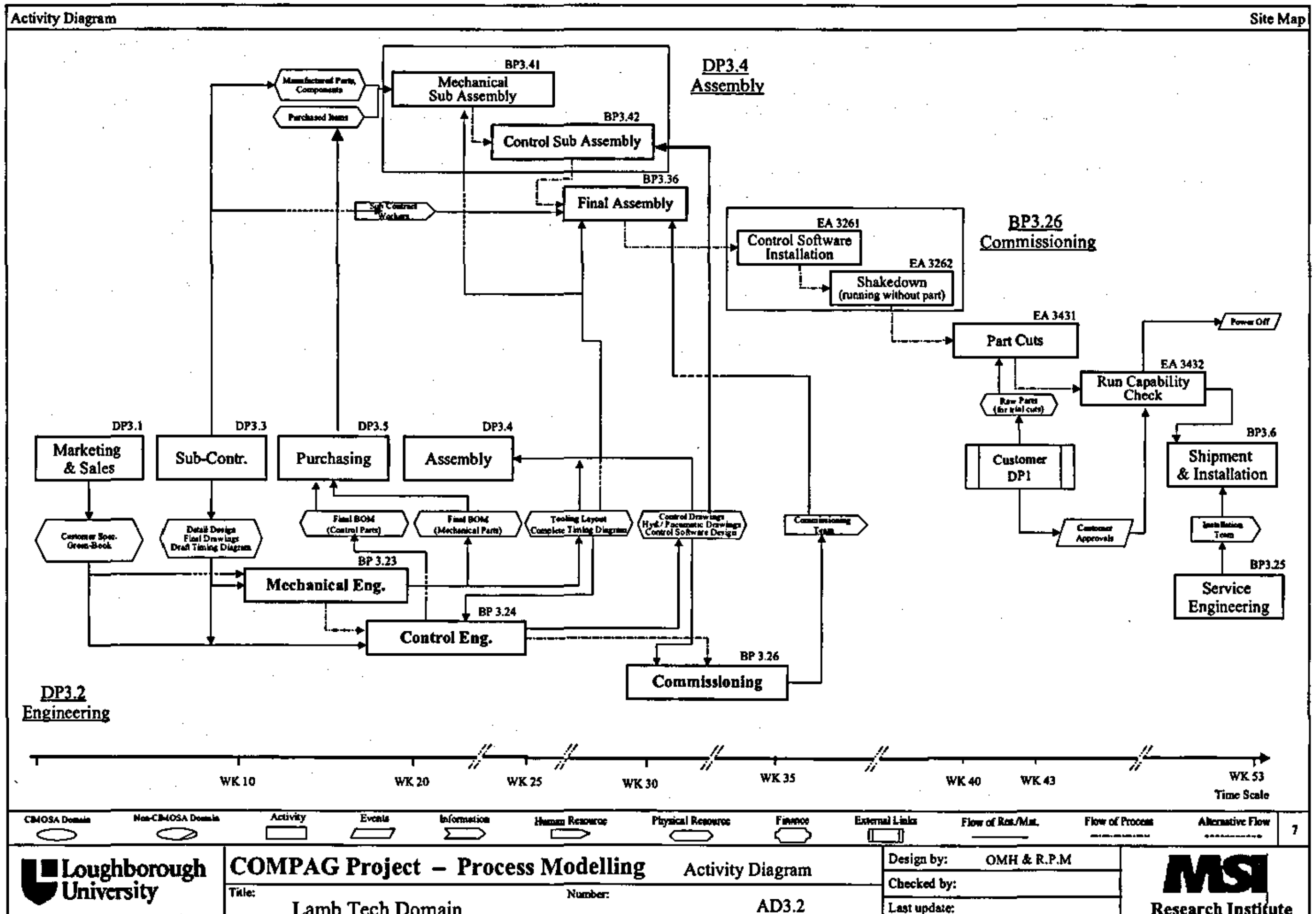


Figure C.5 Activity Diagram (II) for Lamb

Appendix D

Documentation on Johann A. Krause

I) Understanding Processes in Krause

Krause has thirteen milestones for each project, which is shown in Figure D.1. The flowchart in Figure D.1 shows the process after Krause receives a contract. Internally, Krause will organise a team of 10 people to manage the project. There will be a project leader to coordinate the project and a representative from different departments, including mechanical design, control design, manufacturing and assembly. The design work starts with the design of the pallet (i.e. a carrier for the parts), followed by the layout of the assembly line, the design of the conveyor system and then the design of the assembly machine. At each stage, customer approval is required before proceeding to the next stage. When the mechanical design of the machine is completed, a meeting will be held and the design is presented to the other departments. The electrical design will begin after the mechanical design and a prototype is produced for the customer approval. The control design begins after prototyping, followed by mechanical installation and electrical installation. Once the machine has been assembled and wired up, the commissioning begins.

The commissioning usually lasts three to four weeks and only one commissioner is able to commission the machine. Krause has a series of test which are conducted internally to ensure that the machine has met the specifications. When the customer has given their approval for the machine, the machine is taken apart and packed for shipment.

According to one of the commissioner in Krause, about 80% of the machine functions are repetitive. At the moment, to reuse these repeated control functions, the engineers have to "cut and paste" the design from previous machine design. However, it depends on whether anyone in the team could recall which project it was. Krause would like to reuse more of these machine functions more systematically.

At the time of the project, Krause was looking into ways to systematically reuse their design and to establish product data management within the organisation. The main idea is to establish a common database which stores all the common information or data required by the different departments. Once the information/data are entered into the database, it will be available to users in different departments within the organisation. This helps to maintain a systematic achieving of information, ensuring data consistency within the organisations. It ensures that everyone involved in a project is operating on the same set of data.

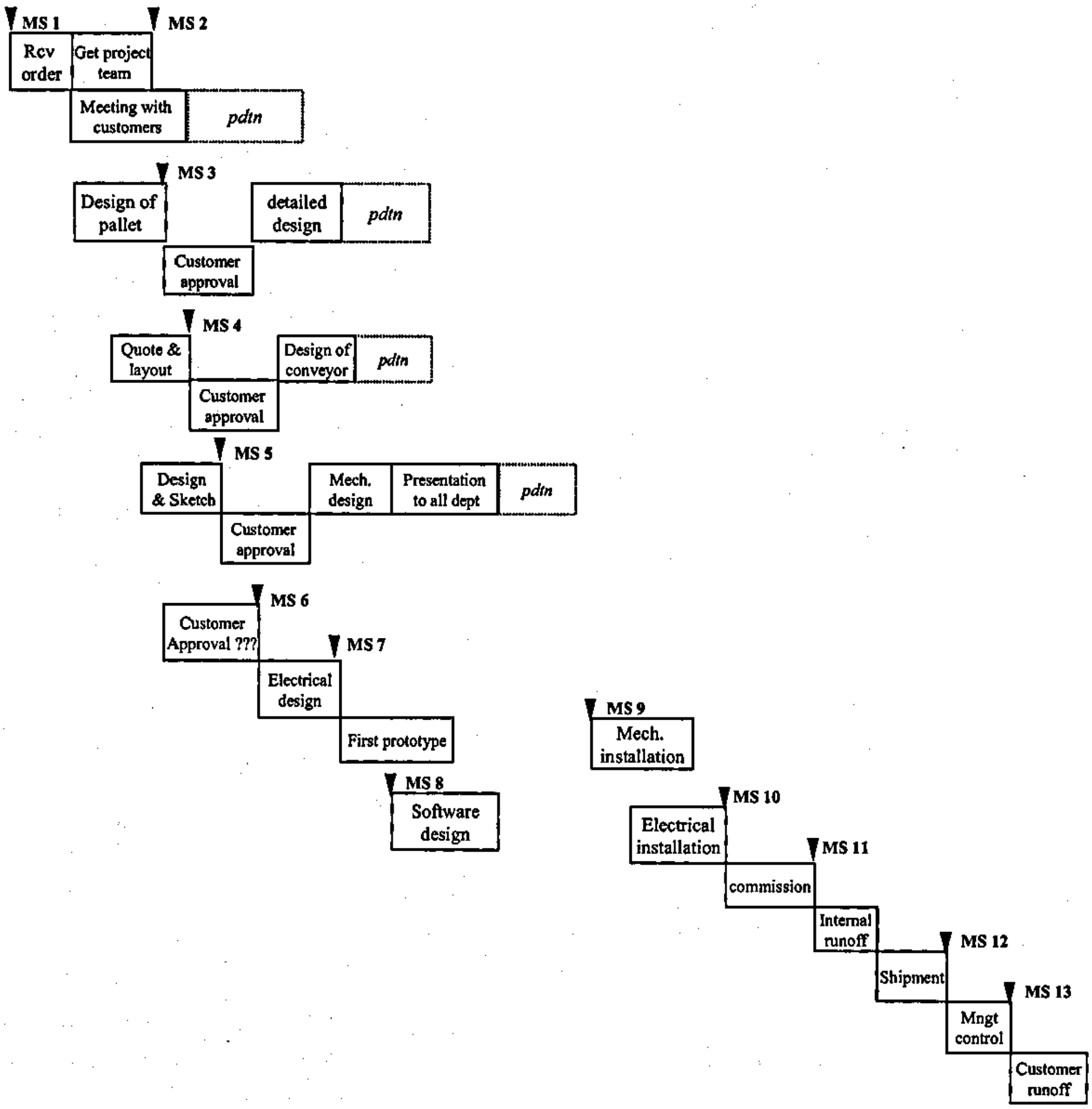


Figure D.1 Krause's Project Milestones

Appendix E

Scenario Testing

I) Proposal for Scenario Testing

Scenario 1: Process Translation

Objectives:

- To assess the level of translation that is required during the process

AS-IS	TO-BE
☞ Paper based documentations requiring translation	☞ The low level control knowledge are encapsulated within a component; the control behaviour is described in a way that can be understood by the user ☞ Control logic available remotely and visually aided by 3D models

Tasks:

- to trace the process of machine design and build
- to count the number of times in which data are translated from one form to another

Matrix :

Factors	Measurements
Effectiveness	The number of translation required

Scenario 2: Modification of Design

Objectives:

- To assess the easiness/complexities of changing the mechanical, electrical and control design
- To assess how the engineering tools will support the changes

AS-IS	TO-BE
Control Logic ☞ Changing control logic requires specialised knowledge such as programmable logic control (PLC) in ladder logic	Control Logic ☞ The low level control knowledge are encapsulated within a component; the control behaviour is described in a way that can be understood by the user

☞ modification of control logic requires the user to go through the logic implemented in ladder logic rung by rung Mechanical ☞ Mechanical engineers will assess the changes for tooling and redesign the mechanical parts	☞ modification of control logic done through changing the interlocks

Task :
- To make a change to the control logic

Role Play:
- control engineer, mechanical engineer, electrical engineer

Observations:
- Observe the procedure required for making the change

Matrix :

Factors	Measurements
i) Level of abstraction	High : Able to implement without the knowledge of internal implementation (i.e. control programming knowledge, such as C++ language, ladder logic, programming instructions)
	Moderate : Able to implement with some knowledge of internal implementation (such as acquire through experience)
	Low : Able to implement with a complete knowledge of internal implementation
ii) Level of experience	High : Complete knowledge of the internal implementation of control system
	Moderate : Do not have the knowledge of the internal implementation of control system but know the causal relationships
	Low : Do not have both the knowledge of the internal implementation and the causal relationship
iii) Complexity	The number of steps that is required

Scenario 3: System Validation / Analysis

Objectives:

- To assess the possibilities of carrying out system testing or validation at an earlier stage.
- To assess the possibilities for sub-system testing or incremental testing.

AS-IS	TO-BE
☞ System verification only during commissioning	☞ System verification at various stages of development using simulation
☞ 3D models constructed separately without the associated control behaviour	☞ Control behaviour of 3D models implemented directly using the real machine logic
☞ Specialised knowledge required to interpret machine behaviour	☞ control behaviour available to all interested parties in a way that the users can easily relate to
☞ Experience based design and implementation	

Task :

- To decompose the system into several sub-systems
- Test each sub-systems individually

Role Play :

- a control engineer

Tools required:

- Krause/Lamb test machines
- PDE
- VRML 3D models

Observations

- Assess the control behaviour of each sub-systems

Matrix :

Factors	Measurement
i) Effectiveness	at which stage of the process is verification possible, the higher the number of stages, the higher the effectiveness
ii) Level of abstraction	High : Able to implement without the knowledge of internal implementation (i.e. control programming knowledge, such as C++ language, ladder logic, programming instructions)

	Moderate : Able to implement with some knowledge of the internal implementation (such as acquire through experience)
	Low : Able to implement with a complete knowledge of internal implementation
iii)Level of experience	High : Complete knowledge of the internal implementation of control system
	Moderate : Do not have the knowledge of the internal implementation of control system but know the causal relationships
	Low : Do not have both the knowledge of the internal implementation and the causal relationship
iv) Accuracy	High : Behaviour of models fits the specification of machine completely
	Average : Behaviour of models fits some of the specification of the machine
	Low : Behaviour of models do not fit the specification of the machine
v) Resources required	Cost : the amount of money required to provide the items
	Time : the amount of time required to provide the items

Scenario 4: Remote Diagnostic/Maintenance

Objectives:

- To assess the efficiency/effectiveness of diagnosing a machine remotely
- To assess the easiness/complexities of monitoring the machine remotely
- To investigate the possibilities of offering multi-perspectives for understanding machine behaviour.

AS-IS	TO-BE
☞ Remote diagnostic possible	☞ The low level control knowledge are encapsulated within a component; the control behaviour is described in a way that can be understood by the user
☞ Control logic available remotely but only in terms of ladder logic	☞ Control logic available remotely and visually aided by 3D models
☞ Remote monitoring possible but without visual aid	☞ Remote monitoring possible and visually aided by 3D models.

Tasks:

Task I

- to inject error onto the machine
- to diagnose the machine remotely using different tools (telephone, video camera, PDE tools)

Task II

- To control the machine from remote location
- To record the machine operation for a period of time
- To replay the recorded machine operation

Role Play :

- operator
- control engineer

Tools:

- Ford Test Rig
- PDE

Observations:

Task I

- To note the process of recovery

Task II

- To note the process of remote controlling
- To note the process of replaying the machine operation

Matrix:

Factors	Measurements
i) Efficiency	Time taken to complete the task
ii) Resources	Cost : the amount of money required to provide the items
	Time : the amount of time required to provide the items

Scenario 5: System Manipulation

Objectives:

- to test the ease of manipulating the system to a desired state
- to assess the level of knowledge required to assess the ability to manipulate the system

Tasks:

- To inject errors to the machine while it is running

Role Play :

- a control engineer / commissioner

Tools required:
- Ford Test rig or Krause/Lamb test machines

Observations:
- Observe the procedures required to recover the machine

Matrix:

Factors	Measurements
i) Level of abstraction	High : Able to implement without the knowledge of internal implementation (i.e. control programming knowledge, such as C++ language, ladder logic, programming instructions)
	Moderate : Able to implement with some knowledge of internal implementation (such as acquire through experience)
	Low : Able to implement with a complete knowledge of internal implementation
ii) Level of experience	High : Complete knowledge of the internal implementation of control system
	Moderate : Do not have the knowledge of the internal implementation of control system but know the causal relationships
	Low : Do not have both the knowledge of the internal implementation and the causal relationship
iii)Complexity	The number of steps that is required

Scenario 6: System Design

Objectives:
- To assess the process of designing and building a machine using the COMPAG tools
- To assess the time difference in using CBA

Tasks :
- to specify, design and build the mechanical system of a machine
- to specify, design and build the control system of the machine
- to specify, design and build the electrical system of the machine

Role Play:
- a mechanical engineering,

- a electrical engineer,
- a control engineer.

Tools required:
- for control, PDE

Matrix:

Factors	Measurements
i) Level of abstraction	High : Able to implement without the knowledge of internal implementation (i.e. control programming knowledge, such as C++ language, ladder logic, programming instructions)
	Moderate : Able to implement with some knowledge of internal implementation (such as acquire through experience)
	Low : Able to implement with a complete knowledge of internal implementation
ii) Level of experience	High : Complete knowledge of the internal implementation of control system
	Moderate : Do not have the knowledge of the internal implementation of control system but know the causal relationships
	Low : Do not have both the knowledge of the internal implementation and the causal relationship
iii)Complexity	The number of steps that is required
iv) Resources	Cost : the amount of money required to provide the items
	Time : the amount of time required to provide the items

II) Diagnostics Process of Machine Builder

Figure E.2 shows the current diagnostic process for machine builders. When the end user encounter problems on the machine, the maintenance engineers will try to establish the problem and the possible solutions. They will try their solutions on the machine and validate the solution. If the problem cannot be solved, they will call the machine builder for help. This is usually done through phone or through exchanges of emails. In most cases, the machine builder's engineers usually rely only on the archived records of the machine logic (i.e. in ladder logic) and try to understand the situation based on the maintenance engineer's description, establishes the problem and the possible solution.

The engineers also relied on their experience in problem diagnostics. With the help of the ladder logic, the engineers will form a mental model of the machine and try to imagine the problem that is occurring. The engineers will suggest the possible solution to the maintenance engineer for the latter to try out on the machine. If the problem still can not be solved, then the machine builder's engineer will have to go on site to solve the problem.

Currently, although the machine builders are able to logon remotely to the end users' machine, they can only access the control software of the machine. They do not have virtual simulation of the machine at real time. Feedback from the machine builders shows that they would like to have visualisation of the machine at real time and check the performance against the control software coding. This will help them establish the problems and derive solutions to the problems.

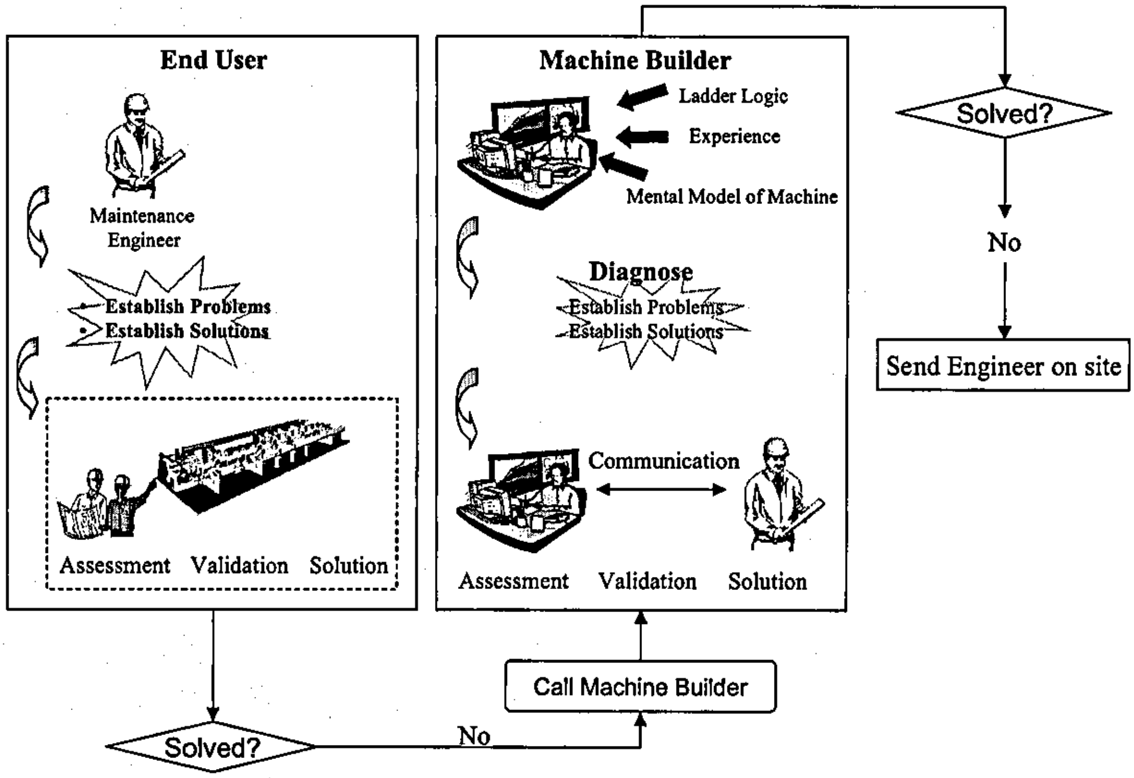


Figure E.1 Diagnostic process under current practice

III) Errors Diagnose / Maintenance

Errors and warnings on the machine are differentiated on the HMI. Errors are problems on the machine which required the operator to attend to it immediately, while warnings are to make the operator aware of the possibilities of error occurring.

Different machine builders have different strategies for error handling. For example, the assembly line machine builder, Krause, classifies errors according to the degree of severity and prioritises the error messages. Severe errors are those which resulted in the machine being out of auto-cycle and require the system to be reset. These errors are given the highest priority and will only clear after the system has been reset. Errors which are less severe, the message on the HMI will clear once the problem has been dealt with. For metal cutting machine builder, Lamb Technicon, critical errors will result in the stopping of the machine. For errors which are less severe, the machine will stop at the end of a cycle or the zone which it is in.

Very often, errors which are not severe is often taken care of by the operator after resetting the machine and will not affect the operation of the machine. The problem may not be severe, but it will affect the overall performance of the machine in the long run. Errors and warnings are important to the maintenance engineer, who needs to keep the machine in good "health", but it requires a lot of discipline from the operator to log down the detailed operation on the machines. Figure E.2 summaries the current practice on the shopfloor as described above.

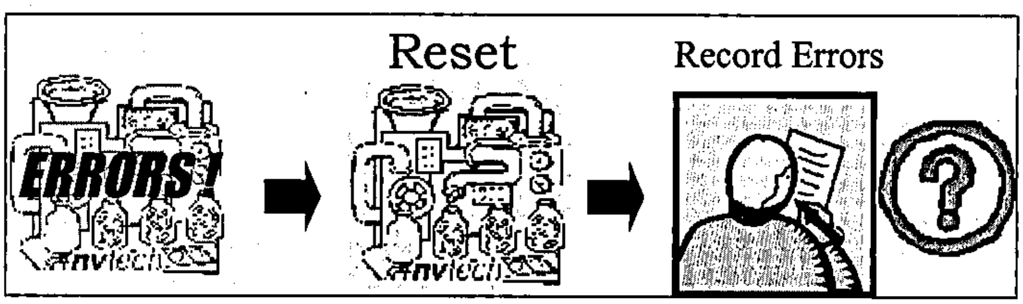


Figure E.2 Error diagnostics under current practice

End users are constantly finding ways to cut down costs and reducing inventory is one of it. Increasingly, they are requesting more information from the machine builders on the failure rate of the components and the machine. They would like to delay the purchasing of the spare components to a later date. They also would like to reduce the frequency of machine maintenance because every maintenance implies machine down time. To provide such information to the end users, the machine builders obviously rely on feedback and the detailed documentation of machine operation from the end users. As mentioned above, it requires discipline from the end users to log down detailed operation on the machines and send them to the machine builders. The process of keeping a good log, regrettably, does not work out very well.

The engineering supporting tools in the COMPAG environment can be used for the maintenance of the machine. The machine builders can logon remotely to the end user's machine and download the operation log of the machine. With an intelligent component which is capable of performing self-diagnosis, such as keeping track on the number of operations it had performed and number of failures, the machine builder can access these information on the components and determine if the components required a replacement. For the end users, such facilities will also enable their own maintenance engineers to perform regular check on the machine, too. The engineers can download the information from the machine, replay the operation using the VRML based 3D model, retrieve the STD or timing diagram and check the operation against the STD and timing diagram. The scenario using the COMPAG tool is illustrated in Figure E.3.

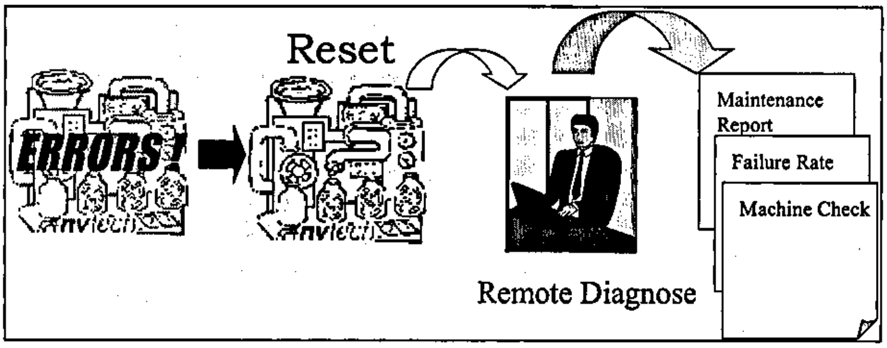


Figure E.3 Using the COMPAG tools for remote maintenance

Appendix F

Publication

1) Ong, M. H., Vera, D., West, A. A., and Harrison, R., 2002, "Human Factors Issues in the Application of a Novel Process Description Environment for Machine Design and Control Developed Under the Foresight Vehicle Programme," Proceedings of the Society of Automotive Engineers World Congress and Exposition, Paper 2002-1-0466, Foresight Vehicle Technology: Supply Chain & Manufacturing/Machine Manufacturing Control (Part C&D), Detroit, USA.

2) Monfared, R.P., Ong, M.H., West, A. A., and Harrison, R, 2004, "Application of Process Modelling and Simulation to Assess the Impacts of Process Change in Manufacturing Systems", NAMRC XXXI, Hamilton, Ontario, Canada.

3) Ong, M. H., Monfared, R. P., Lee, S. M., West, A. A., and Harrison, R., 2004, "Evaluating the Implementation of the Component-Based System in the Automotive Sector", Proceedings of 7th Biennial ASME Conference: Engineering Systems Design and Analysis (ESDA 2004), Manchester, United Kingdom, ASME.

4) Lee, S. M, Ong, M.H., West, A.A. and Harrison, R, 2004, "The Need for a Component-Based Approach to Automation System for Agile Manufacturing", Proceedings of 7th Biennial ASME Conference: Engineering Systems Design and Analysis (ESDA 2004), Manchester, United Kingdom, ASME.

5) Ong, M.H., Lee, S.M., West, A. A., and Harrison, R, 2004, "Evaluating the Use of Multimedia Tool in Remote Maintenance of Production Machinery in the Automotive Sector", 2004 IEEE Conference on Robotics, Automation and Mechatronics, Singapore.

6) Ong, M.H., West, A. A., and Harrison R., 2004, "A Structured Approach to Evaluating the Impact of Implementing the Component-Based System in the Automotive Sector", submitted to International Journal of Production Research.