Distributed and collaborative product development and manufacturing knowledge management

Cheung, Wai Ming

Durham University

SCHOOL OF ENGINEERING

WAI MING CHEUNG

Distributed and Collaborative Product Development and Manufacturing Knowledge Management

Ph. D. Thesis

2007

Supervisors: Prof. P. G. Maropoulos

Dr P.C. Matthews

This thesis is submitted to the University of Durham in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

DECLARATION

I confirm that no part of the material offered in this thesis has previously been submitted by me for a degree in this or any other university. This thesis presents the author's own work, except where appropriately acknowledged citations are given. Where material has been generated through joint work, my independent contribution has been clearly indicated.

COPYRIGHT NOTICE

W**ai **M**ing **CHEUNG
School of Engineering
University of Durham

ACKNOWLEDGEMENTS

I wish to express my sincere thanks to my supervisors Prof. Paul G. Maropoulos and Dr Peter C. Matthews. This research is based on the ideas of supporting early design conceptualisation and manufacturing scenarios by Prof. Paul G. Maropoulos and Dr James X. Gao, both of whom formulated much of the early thinking on the research hypothesis. I am also deeply thankful to my colleague from Cranfield University, Hayder Aziz, who came up with the idea of using *open source* solutions in this research.

I owe a debt of gratitude to many other people in the School of Engineering at the University of Durham. In particular, I acknowledge the significant contribution of Dr David G. Bramall and Kevin R. McKay, both of whom developed the CAPABLE Space System which later was used and applied in this research. In particular, David, who has given me support and advice in facilitating the CAPABLE system. I also acknowledged Steven Kelly from Siemens (UK) Ltd, for his valuable time to read the thesis.

The industrial perspective plays an important part in this thesis: I am indebted to the contributions made by various individuals at the main industrial sponsors, in particular, Guy Bursell and Steve Moore from Mabey & Johnson Ltd. In addition, this project would not have been possible without the financial support of the Engineering and Physical Sciences Research Council (EPSRC), UK.

Finally, a special thanks is also due to the *river bank* of Durham Castle right across the end of *Findchdale Priory*, here I have spent countless hours of jogging, running and the solitarily of contemplating ... *keep smiling through life's ups and downs!*

ABSTRACT

A major challenge facing industry nowadays is the adoption of current information sharing and collaboration technologies for early design and manufacturing phases to enable collaboration for product development through information distribution. Furthermore, product development process displays ever growing complexity and dynamic behaviour due to both increasing product complexity and distributive and collaborative engineering demands. In order to overcome these issues, advanced strategic corporate alliances must share knowledge, expertise and resources in an increasingly competitive global market.

The principal hypothesis of the research is that, at present there is a disconnection between the early stages of communicating design concepts and potential manufacturing scenarios which could be bridged by using internet-based Product Data Management (PDM) systems with techniques and methods for design conceptualisation, aggregate factory modelling and manufacturing knowledge management. The main objective of this research is therefore to investigate and develop methods for the effective management of the internet-based process of communicating new product requirements and manufacturing performance evaluations. In particular, the investigation is focused on the early stages of product development throughout the product life cycle using PDM, Enterprise Resource Planning systems and Web-based technologies. The tools used to explore the feasibility are the utilisation of Web-centric supporting technologies such as eXtensible Markup Language, Resource Description Framework and ontologies for manufacturing knowledge management. The Unified Modelling Language and Object-Oriented based Java Programming Language are used to further develop and facilitate an early process planning evaluation system. In addition, a new framework using Peer-to-Peer technology as a subset of the product development integration architecture to support smaller companies and large corporations has also been developed.

This thesis presents the contributions and the development of novel methods which include:

1. The aggregate manufacturing models,

2. New methods in relation to knowledge management of design and manufacturing,
3. A client/server product development integration architecture, and
4. A decentralised production network for smaller and larger companies using 'open source' solutions.

These will facilitate the communication of early design and product development within a distributed and collaborative environment.

Two case studies are presented to verify the demonstrations. The first case study replicates a centralised client/server environment supporting the design of steel panel bridges for rapid assembly on-site. The second case study is focused on creating a virtual enterprise collaboration to compare the applications of a centralised PDM and decentralised open source solutions. Overall results have indicated that:

1. The opportunity of early collaboration in product design can be maximised, prevent poor decisions, enable the design to be right first time, and
2. Elimination of bottlenecks in bandwidth and resources, reduce centralised administration cost and empowering of collaborators within networks to control the knowledge they create.

Therefore, with proper technologies, methods and techniques to share knowledge, expertise and resources, can enhanced the three critical factors in product development, namely: reduction of cost, time-to-market and quality of product.

ABBREVIATIONS

2D	2-Dimensional
3D	3-Dimensional
AI	Artificial Intelligence
AP	STEP Application Protocol e.g. AP214, AP224
API	Application Programming Interface
APS	Advanced Planning and Scheduling
B2B	Business-to-Business
ASP	Application Service Provider
BoM	Bill-of-Materials
C++	An object oriented programming language
CAAP	Computer Aided Assembly Planning
CAD	Computer Aided Design
CAM	Computer Aided Manufacture
CAPABLE	Concurrent Assembly and Process Assessments BLock for Engineering manufacture
CAPP	Computer Aided Process Planning
CBL	Common Business Library
CDW	Cost Data Warehouse
CE	Concurrent Engineering
CLIPS	C-Lisp language to define knowledge bases and expert system rules
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
CPC	Collaborative Product Commerce (an extension of PDM for inter-company data sharing)
CPD	Collaborative Product Development (an extension of CPC for product development)
CRM	Customer Relations Management
CRP	Capacity Resource Planning
CSG	Constructive Solid Geometry

cXML	commerce eXtensible Markup Language
DFA	Design for Assembly
DFM	Design for Manufacture
DFMA	Design for Manufacture and Assembly
DfX	Design for X (can be assembly, manufacture, reliability, maintainability)
DMU	Digital Mock-U
DTD	Data Type Definition (define XML tags)
DTI	The Department of Trade & Industry
EDM	Engineering Data Management
EDM	Engineering Document Management
ERP	Enterprise Resource Planning
EXPRESS	programming language used to define STEP complaint files
GT	Group Technology
GUI	Graphical User Interface
IGES	Initial Graphics Exchange Specification
HTML	HyperText Mark-up Language
IDE	Integrated Development Environment
ISO	International Standards Organisation
JDK	Java Development Kit
JXTA	JuxtaPose
KBS	Knowledge-Based System
KE	Knowledge Engineering
KM	Knowledge Management
KQML	Knowledge Query Manipulation Language
LDAP	Lightweight Directory Access Protocol
MCAD	Mechanical CAD
MBOM	Manufacturing Bill-of-Materials
MRP	Materials Resource Planning
MRPII	Manufacturing Resource Planning
OBI	Open Buying on the Internet
ODBC	Open Database Connectivity

OEM	Original Equipment Manufacturer
OLE	Object Linking and Embedding
OMG	Object Management Group
OO	Object Oriented (a programming paradigm)
OWL	Web Ontology Language
P2P	Peer-to-Peer
PDM	Product Data Management
PICS	Platform for Internet Content Selection
PIM	Product Information Management
PLM	Product Lifecycle Management
PSL	Process Specification Language
RAD	Rapid Application Development
RCCP	Rough-Cut Capacity Planning
RDF	Resource Description Framework
RMI	Remote Method Invocation (Java Distributed Objects)
ROI	Return On Investment
SAX	Simple API for XML parsing
SDAI	Step Data Access Interface
SGML	Standard Generalized Markup Language
SME	Small and Medium-sized Enterprise
STEP	Standard for the Exchange of Product Model Data
TBPM	Task Based Process Management
TCP/IP	Transmission Control Protocol/Internet Protocol
TIM	Technical Information Management
TQM	Total Quality Management
UML	Unified Modelling Language
VE	Virtual Enterprise
VPR	Virtual patient record
VR	Virtual Reality
VRML	Virtual Reality Modelling Language
WWW	World Wide Web
WfMC	Workflow Management Coalition
XMI	XML Meta data Interchange

XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language

Chapter 1

INTRODUCTION

1. INTRODUCTION

1.1 INDUSTRIAL PROBLEMS AND SOLUTIONS

The manufacturing industry faces many challenges such as reducing time-to-market and cutting costs that directly affect the company and the life cycle of a product. According to Maropoulos and Gao (2000), the key industrial problem for modern manufacturers is the lack of collaboration at the early stages of distributed product development due to the following reasons:

- The lack of techniques for the rapid translation of early design ideas into an analysable form, and
- The lack of meaningful manufacturing knowledge in the feedback evaluation process

The industrial impact of the above problems is felt in many areas, such as vital decision-making especially in the areas of product configuration and responsiveness to changing markets and meeting customer demands. A further problem is the inability to make full use of e-business technology. Another huge problem the industry faces is meeting individual customer requirements such as *engineered-to-order*. The product's life cycle is increasingly shortened, volume reduced and customisation increases.

Therefore, it is important to interact with the client to meet product definition during the earliest concept stage of product development. Hence, it is imperative that the Original Equipment Manufacturers (OEM), suppliers, vendors and customers can interact effectively and make a conceptual design for the product that is manufacturable and cost effective. The way to achieve this is by effective communication within the enterprise from conceptual product design and manufacturing. Communication is vital for decision making and support in the product development process, thus, this is one of the main factors contributing to product quality, cost and delivery.

Solutions to these problems are proposed and described below. In order to meet the development in a distributive and collaborative environment the research work is divided between the University of Durham and Cranfield University. The two major building blocks are in-place as follows:

1. The novel aspects of the research work developed at Durham are:
    - The ability to easily create, modify and utilise design and manufacturing knowledge during the early design phase.
    - To create assembly plans for the components and evaluate the potential viability using assembly planning tool and interface with Resource Planning tool.
    - Product Data Management (PDM), which holds all design data and meta-data as well as enabling version control of design iterations and access to shared work area for the team members.
    - Secure data communications technologies to allow data to flow between team members and the central repository.

2. The novel aspects of the research work developed at Cranfield are:
    - Computer Aided Design (CAD) environment to generate conceptual designs and allow the visualisation of product design for non-technical members and for design evaluation.
    - Viewer technologies to allow real-time collaboration of geographically disparate groups, allowing them the ability to mark-up designs and

recommend changes. Also, to allow persons without access to CAD software to view the models.

- Interface between the CAD software and Process Planning software in order to assess the potential manufacturability of a design from the conceptual phase. Interface with Resource Planning tools will also forecast the potential costs of the design from an early stage of the project.
- The ability to pass the conceptual design as a "rough sketch" directly into the CAD tool for detailed design.

As such, the following sections will discuss the research hypothesis, the methods and the activities to be undertaken in order to meet the proposal.

1.2 RESEARCH HYPOTHESIS

The principal hypothesis of the research is that at present there is a disconnection between the early stages of communicating design concepts and potential manufacturing scenarios. However, this discontinuity could be bridged by using internet-based PDM systems with techniques for design conceptualisation and aggregate factory modelling and manufacturing knowledge management. The main objective of this research is therefore to investigate methods for the effective management of the internet-based process of communicating new product requirements and manufacturing performance evaluations. The investigations will focus on the critical early stages of product development throughout the product life cycle using PDM and related Web-based technologies. The potential industrial benefits will be the demonstration of increased front-end responsiveness and improved quality of design and production decisions that will allow supply chain companies to take full advantage of emerging e-business models. The research is particularly relevant to *manufacturing environments with small batch sizes, medium to high product complexity, increasing product variety and decreasing product life cycles.* Under such circumstances, the efficient management of product and process knowledge, from the early stages of design, should result in reduced product development lead times and increased production efficiency.

1.3 THE MAIN METHODS AND ACTIVITIES OF THE RESEARCH

1.3.1 Research Methodology

The technical methodology is illustrated in Figure 1.1 that shows the overall architecture of the proposed project in relation to product life cycle:

- From conceptual design (aggregate product modelling) via aggregate manufacturing knowledge modelling to rough-cut capacity planning and,
- From detail design (Computer Aided Design - CAD) to capacity requirements (ERP).
- The discontinuity of early design concepts and potential manufacturing scenarios could be bridged by the synthesis of internet-based PDM systems with novel techniques for design conceptualization via knowledge management (Knowledge-Based System - KBS) and aggregate factory modelling (the CAPABLE system (Bramall *et al.* 2003)).
- The proposal also sets out to explore the adoption of a client/server centralised and P2P decentralised production networks.

The illustration highlights the individual research areas between Durham and Cranfield Universities. The research undertaken at the University of Durham is:

- *Conceptual Design* - development of the Aggregate Product Modelling including the Aggregate Process and Factory Models of the CAPABLE system.
- *Manufacturing Knowledge Modelling* - for capturing implicit and explicit knowledge in design and manufacturing.
- *Integration Architecture* - to create a methodology of an integrated environment based on the applications of PDM, ERP systems and Web-based technology.

The research undertaken by Cranfield University is:

- *Detail Design* - the development of a STEP Modeller (STandard for the Exchange of Product model data) in AP224 (Application Protocol).
- *Manufacturing Knowledge Modelling* - for capturing design geometric data and design rules.
- *The application of P2P method* - the author of the University of Durham has made significant contribution to several elements such as distributed indexing,

security issues, the use of workflows and refining the connectivity of the infrastructure and testing of P2P.

The discussion in this thesis will focus on the research carried out at the University of Durham. The research methods include literature review of related work and technologies in collaborative product development, PDM/web technology assessment via desk-based analysis of vendor literature, evaluation of technical demonstrations and action research in collaboration with industry. This will be followed by the definition of new techniques and the prototyping of selected computer functions in a pilot technology demonstrator. One of the key issues the proposed system seeks to address is the lack of intuitive collaboration at the early stages of product development within an enterprise, by including meaningful manufacturing knowledge and constraints in the product development evaluation process.

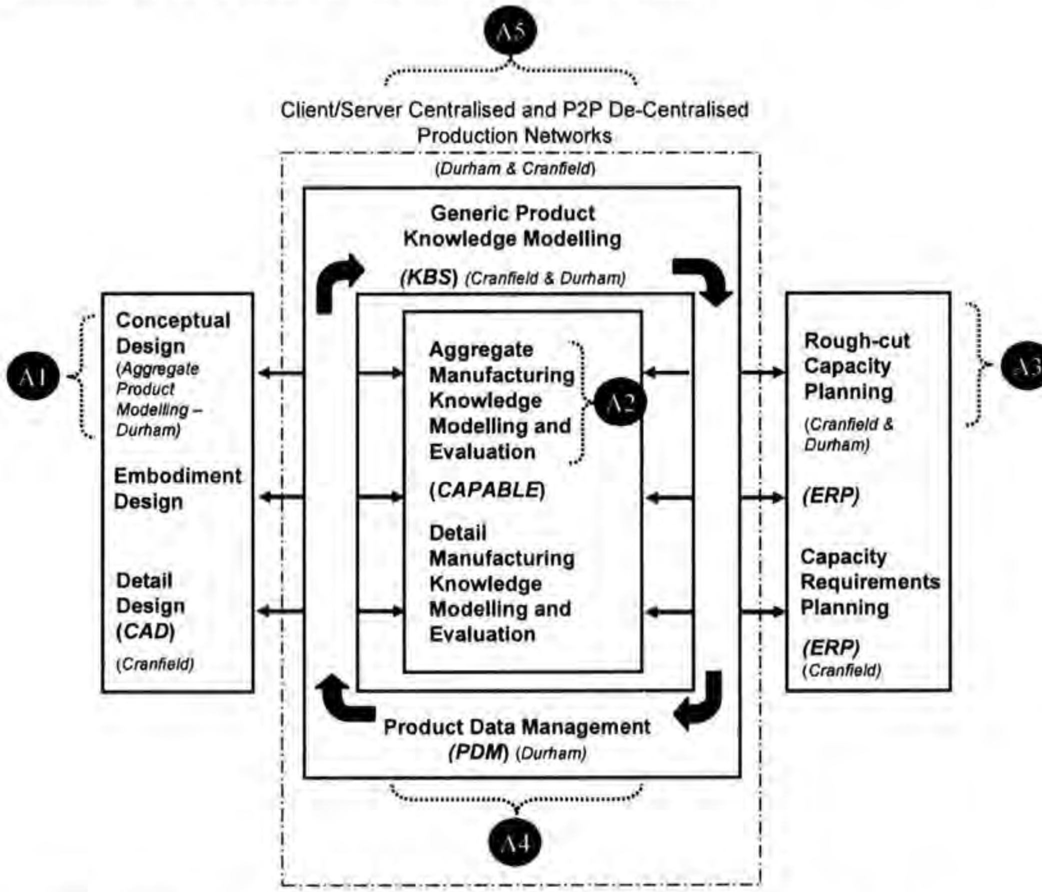


Figure 1.1: The Main Elements of the proposed Distributed and Collaborative Product Development and Manufacturing Knowledge Management Environment (Maropoulos and Gao 2000).

One important aspect of this research is to present a means of creating a single knowledge domain for manufacturing similar to Goasdoue's (1999) approach. The method of sharing knowledge in this work makes use of existing technology such as a PDM which allows the same information to be shared heterogeneously within a collaborative and distributed secure environment. Another aspect of this work adopts an approach similar to Moench's (2003) work which enables users to retrieve knowledge for various applications. This method of knowledge retrieval makes use of the application of XML Parser technology which enables the XML-formatted knowledge to be shared by other third party applications.

The scope of the research is restricted to processes for which there are already available modelling methods and will be centred on the PDM and manufacturing knowledge modelling aspects. With regard to design and ERP, only a small number of essential functions will be used to generate the product model and assess capacity implications respectively. The PDM functions will provide the "communications wrapper" linking design with manufacturing and capacity planning. The term "communications wrapper" is used to denote a coherent layer of proprietary functions and data management procedures that conform to international standards, such as XML, so that it can be used to facilitate and manage the process of linking design and manufacturing functions and systems.

The leading edge Web-centric PDM system Windchill has been chosen for modelling product knowledge and controlling design and manufacturing processes (workflow) at all stages of design. Product information will be generated by the CAD system, Pro-Engineer. CAPABLE, a prototype aggregate process planning system will be used for converting early design specifications into production routings with rough-cut predictions of times and costs. The ERP system Compiere will be used to generate capacity plans using predicted demand for routes generated by CAPABLE aggregate routes, corresponding to early designs. This is the first time that a capacity based assessment of non-linear aggregate routes has been attempted with a view of choosing routes that will improve factory loadings. The selected tools and the developed methodology will ensure that this project is in the forefront of relevant research internationally.

1.3.2 The Main Objectives

The aim of the research described in this thesis was to propose a product development integration architecture to support early design stages and manufacturing evaluations to:

1. rapid translate early design ideas into an analysable form, and
2. include meaningful manufacturing knowledge in the feedback evaluation process.

To confirm this view the following research activities were adopted by the University of Durham:

1. A traditional review of the literature was undertaken. This focused was on journals, trade magazines, books and work being undertaken in the consulting area.
2. To assess the current approaches adopted by industry through:
    a. an assessment exercise carried out in PDM commercial systems, and
    b. a survey undertaken on how industrial companies have implemented PDM.
3. To translate early design concepts/ideas into appropriate, tentative product configurations for detailing/embodying and measuring resulting manufacturing performance as shown in Figure 1.1 (A1). This includes the development of Aggregate Manufacturing Modelling in product, process and factory resource in using Unify Modelling Language (UML) and JAVA programming techniques.
4. To investigate whether by linking the design intent translation and manufacturing knowledge methods with web-centric PDM systems to demonstrate the effective generation and communication of design ideas and manufacturing evaluations from the critical early development stages.
    a. A novel organisational knowledge framework using ontology to represent design, manufacturing and business process has been defined and developed as shown in Figure 1.1 (A2).

   b. The methods for integrating design with PDM and ERP, including the development of an advanced integration architecture using linking methods of Workflow Activity Task Controller (WATC) and XML Parser as shown in Figure 1.1 (A3 and A4)

   c. Define and research additional work on P2P to create a hybrid integration methodology as shown in Figure 1.1 (A5).

5. To evaluate the potential benefits by testing the developed methods and prototype systems in two exemplars:

   a. replicates a centralised client/server environment based upon steel bridge panel for rapid assembly on-site, and

   b. uses a vehicle-door-latch product and the application of PDM and decentralised peer-to-peer network for smaller and larger companies collaboration .

1.4 INTRODUCTORY REMARKS

This thesis describes the work undertaken during the research into the development of a collaborative and manufacturing knowledge distributed environment. The fundamental objectives and various aspects of the research proposal and activities have been described. The research work was a collaboration project with Cranfield University and industrial partners, Arvin Meritor Ltd and Mabey & Johnson Ltd. The underlying research was carried out under a grant from the UK Engineering and Physical Sciences Research Council (EPSRC) (GR/R26757/01) and (GR/R R3514801/01. The technologies and software tools used to engage in the research are quite diverse including the utilisation of Web-centric supporting technologies such as eXtensible Markup Language (XML) and Resource Description Framework (RDF), the Unified Modelling Language (UML), Java Programming Language, the CAPABLE system and a commercial Product Data Development (PDM) system. Other technology the research has deployed includes the *open source* software tools such as Enterprise Resource Planning (ERP) system, a knowledge management editor and Peer-to-Peer (P2P) networking.

This thesis sets out the requirements for the research at Durham and provides a discussion of aggregate process planning concepts and the assessments of enterprise integration technologies. In addition the thesis also describes the modifications and further developments of the CAPABLE/Space System (Concurrent Assembly and Process Assessments BLock for Engineering manufacture) (Maropoulos *et al.* 1998; Bramall *et al.* 2003; McKay *et al.* 2003) to accommodate and meet the needs of the collaborating companies' product specifications and manufacturing facilities. A novel organisational knowledge framework using ontology to represent design, manufacturing and business process has been defined, in particular the development of 'manufacturing know-how' to enhance design decisions. An integration architecture for product development has been developed to facilitate bridging the gap between the application of PDM, Enterprise Resource Planning (ERP), Web-based technologies and manufacturing and design domains. The integration of these disparate technologies will be of benefit to collaborative work teams, to enable them to interface and develop, review, analyse and reuse engineering and manufacturing knowledge and models within the manufacturing enterprise. In addition a new framework using Peer-to-Peer (P2P) technology as a subset of the product development integration architecture to support smaller companies and large cooperation has also been developed. The aim of this additional application is to test the advantages of adopting combinations of centralised and de-centralised production networks.

1.5 STRUCTURE OF THE THESIS OVERVIEW

This thesis is organised into 9 chapters which are divided into 4 main parts as shown in Figure 1.2. Part 1 covers the research hypothesis and theoretical aspects of the literature review. Part 2 covers the building of the theoretical framework of the product development integration architecture and its associated components. In part 3, the thesis covers data collection for the case studies. Part 4 covers the resulting conclusions and future works of this research. The appendices contain further information related to certain chapters of the thesis.

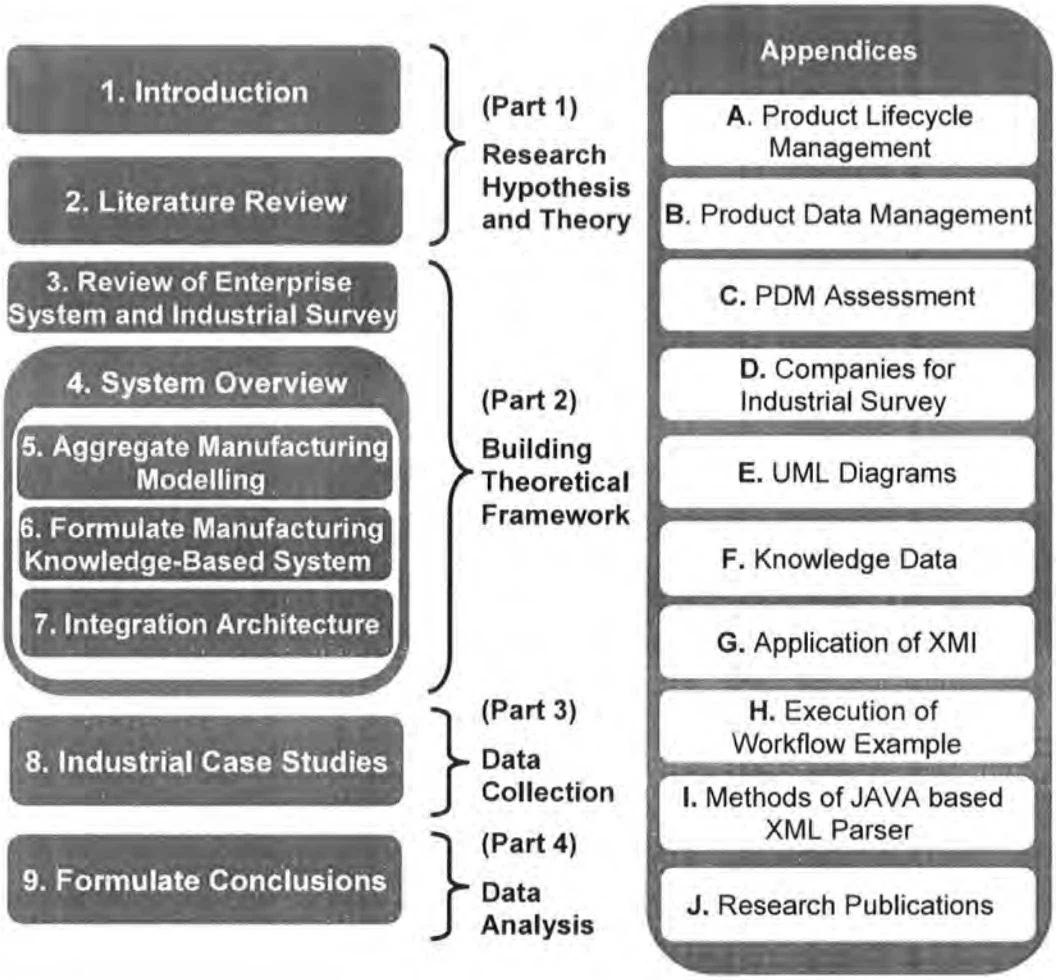


Figure 1.2: Structure of the Thesis

Chapter 1 Introduces the research background, the methodology and activities.

Chapter 2 Presents the literature review covering a wide range of existing work and technologies.

Chapter 3 Review of the enterprise system in PDM and industrial survey which are a part of the software component used in Chapter 7

Chapter 4 Describes the various software components and scope of industrial applications

Chapter 5 Describes the implementation and additional capabilities of the CAPABLE System for early process planning evaluations to be used in the integration environment.

Chapter 6 Describes the framework and the structure of an ontology-based Organisational Knowledge-based System, the criterion and elements such as axioms, constraints and Web-based technology needed to create a knowledge-based system to be used in the integration architecture with the PDM and CAPABLE Systems as described in chapters 5 and 7 respectively.

Chapter 7 Describes the implementation and methodologies needed to create the integration environment.

Chapter 8 Displays the results based on the two case studies.

Chapter 9 Discusses and concludes the founding of this research with further recommendations.

Appendices A and B describe the core areas and the functionalities of PLM and PDM systems. Appendix C shows the PDM providers and their offerings. Appendix D shows a list of companies in the industrial survey. Appendix E illustrates the additional capabilities of product, process and factory resource models. Appendix F illustrates the data collected for the Organisational knowledge-based system. Appendix G shows how the application of XML Meta data Interchange (XMI) to create the ontology.

Appendix H displays an example using workflows function of the PDM system for the novel methodology, Workflow Activity Task Controller (WATC), to demonstrate the *time based integration wrapper* concept. Appendix I depicts the methods of creating a JAVA based XML Parser. Appendix J shows a list of *twenty-one* publications related to this research, *five* international journals, *two* book chapters and *fourteen* conference papers. Of these, *eleven* were principally written by this author. Over this research period, *fourteen* papers relating to the project have been presented at various international conferences.

Chapter 2

LITERATURE REVIEW

2.1 INTRODUCTION

As stated and discussed in Chapter 1, the overall objective of this research is to bridge the discontinuity of early stages of communicating design concepts and potential manufacturing scenarios. The methodologies to meet the requirements are the application of internet-based enterprise integration systems with techniques in relation to design conceptualisation, aggregate manufacturing modelling and Web-based technologies for manufacturing knowledge management. This chapter therefore focuses on describing the background and review of the relevant techniques and technologies to be used throughout the research. In general, this chapter is categorised into the following sections:

- Related work in collaborative product development,
- The enterprise integration technologies,
- The process planning and its associate modelling components,
- Knowledge management and Web-based knowledge management methodologies and its associate technologies,
- Integration of Applications,
- Applications of workflows, and
- The Peer-to-Peer networks.

This literature describes related work, the technologies, techniques and the background of the approaches which in some cases were the driving force and inspirations behind this research work. The roles of enterprise integration systems and the applications of Web-based and workflows technologies are discussed. Important concepts in relation to aggregate process planning and knowledge management are examined. The motivation and growing interest for implementing open source Peer-to-Peer decentralised network in an inter-enterprise collaboration in product development is also discussed. The deployment of the technologies and different software systems is to formulate novel methodologies in order to create a *product development integration architecture* to demonstrate the benefits and verify the research hypothesis.

2.2 RELATED WORK IN COLLABORATIVE PRODUCT DEVELOPMENT

With the advances in information and web-based technologies over the last decade, there is a shift of research focused on product development. The most recent research is focused on collaborative design and information distribution to the right place, to the right people and at the right time. It aims to meet the increasing demands of the globally co-operative design and outsourcing trend in manufacturing. Recently, a number of research projects have been carried out to support collaborative and distributed solutions from the perspective of CAD, PDM, workflow management, knowledge management and Web-based technologies (Xiao *et al.* 2001; Qiang *et al.* 2001; Xie and Salvendy 2003; Xu and Liu 2003; Li *et al.* 2004a and b; Madhusudan 2005; Rodriguez and Al-Ashaab 2005; Huang *et al.* (2000), (2001) and (2002)).

Xiao *et al.* (2001) developed a Web-based Distributed Production Realization (Web-DPR) system as an infrastructure to support collaborative design and manufacturing. Based on the Java remote method invocation (RMI) mechanism, agents and an event-based mechanism, the functional modules of the systems can be linked and co-ordinated effectively. However, the application is not specifically designed for conceptual design or for passing data on to process planning applications. On the other hand, Qiang *et al.* (2001), developed a collaborative product design support environment based on the Internet. The key aspect of the research is to allow product designers to exchange and share product data and communicate with team members to modify geometry data on particular aspects of the design, and maintain operations consistency in all the distributed cooperative sites on a wide variety of platforms. A limitation of this approach is that the macro operations can only be replayed on workstations using the same CAD software. Similarly, Xie and Salvendy (2003) developed a mechanism to co-ordinate remote members in the process of a collaborative project. The mechanism is able to actively provide the constant awareness and feedback of statuses and activities of members contributing to the whole collaborative task. For example, the information about who the collaborators are, where they are now, and what they are doing. The authors have noted the shortcomings of this prototype system as there is no version control or security features. However there is some other disadvantage relevant to real-world application such as the inability to share data with other CAD users. To address the above limitations, Xu and Liu (2003) developed an architecture utilizing a Web-enabled PDM system in a collaborative design environment. The system is implemented using Microsoft Visual Basic and runs in the Microsoft Windows environment and the internet to allow users on a wide variety of platforms to access the product data. The research however is focused on the detail stage of the design. The authors have noted that the implementation of the system is partial and further research is needed for transforming geometry schema into the object-oriented schema. Visual Basic executables rely upon run time libraries which need to be stored on the client machine which makes it inflexible.

Li *et al.* 2004 (a) developed a client/server framework to enable a dispersed team to accomplish a feature-based design task collaboratively. In this research, the establishment of the distributed design environment is based on RMI. The process of designing a part collaboratively in the environment is server side. The collaborative server can create and manage dynamic sessions which can be accessed by clients to provide a workspace to carry out collaborative design activities. Designers participating in the same session can share the same design model. The authors have noted that there are still some technical problems to be addressed. Firstly, the current information management on the server is a file system-based which can be replaced by a database system. Another issue is the system lacks detailed visualisation information of mechanical parts in order to support web-based collaboration. However, Li *et al.* (2004 (b)) has also developed an Internet-enabled system based on Java, RMI and Web technologies to support collaborative and concurrent engineering design by integrating three functional modules, namely, co-design, Web-based visualisation and manufacturing analysis. In the co-design module, designers are equipped with co-modelling and co-modification facilities to carry out a design task collaboratively. The Web-based visualisation module provides a portal for users to view and analyse a design part conveniently. Manufacturing analysis module can be invoked by users to evaluate and optimise the manufacturing costs and the manufacturability of a design part. This system can be used for a design team geographically distributed to organise a 3D collaborative and concurrent engineering design.

The applications of workflows and knowledge management have been used to support a collaborative product development, for example, the most recent research are Madhusudan (2005) and Rodriguez and Al-Ashaab (2005). Madhusudan (2005) developed an Agent-based Process Coordination (APC) framework for distributed design process management. The approach is to embed autonomous agents in a workflow-based distributed systems infrastructure. The framework utilizes a centralized decision-making and task sharing approach to support design activities. A design process plan is executed by a centralized coordination agent with the help of service agents. However, the research does not state how the data is to be shared across different applications in the downstream processes and whether the software tool works in a real-time collaborative environment. Rodriguez and Al-Ashaab (2005) proposed a knowledge driven collaborative product development (CPD) system architecture. The research is focused on the provision of real-time manufacturing knowledge to support geographically distributed companies in making engineering decisions. The sources of manufacturing knowledge are the manufacturing process, resource capabilities, company experience, technical documents and industrial heuristic knowledge. The architecture developed is modular-based and the manufacturing knowledge model and the product model are implemented as object-oriented databases. The information is accessed using a back-end connectivity CORBA (OMG 2006). However, the authors have stated that there is no real time visualization of the geometry and therefore the design cannot be modified over the internet. Another shortcoming is that the research did not address the problem of how manufacturing knowledge can be represented by a common format to be shared in geographically distributed companies.

Among above authors Huang *et al.* are particularly focused in collaborative product development. Huang *et al.* (2001) developed a Web-based system to manage engineering changes (ECs) in a collaborative product development activity. ECs frequently happen during a design process, and managing the ECs in a Web-based system can facilitate better information sharing, simultaneous data access and more prompt communications among team members. The system can play as a complementary tool to a PDM system to enhance its capability in the management of ECs. Meanwhile, Huang extended the Web-based system to support product design review to support a design chain (Huang and Mak, 2000; Huang 2002). The design review system functions as follows:

- Simulate an on-line central review meeting room equipped with a Virtual Reality Modelling Language (VRML) whiteboard for visualising an on-line design model.
- A review co-ordinator to provide a set of facilities for a project manager to plan the activities and resources involved in the review process.
- A Bill-of-Materials (BoM) explorer to store and share review comments and some relevant documents.

However, the above paradigm does impose a potential hindrance in that the paradigm requires a series of repeatable request-download processes of static HyperText Mark-up Language (HTML) pages and local execution. Under this paradigm, once the download process finished, the server losses control of the relevant HTML pages. Hence, this will cause undesirable result such as the up-to-date information for design changes may not be available to other clients in the collaborative product development activities.

Among all the research discussed however, none of them particularly addressed collaborative product development and information distribution to support the early design stages with disparate technologies and software tools, which will increase the potential industrial benefits of front-end responsiveness, quality of design and production decisions. The combined disparate technologies include knowledge management using ontological techniques supported by PLM, ERP, aggregate manufacturing modelling, workflow management and XML data exchange format. One of the key aspects within the research is the introduction of a product development architecture to integrate the disparate technologies. Another research area introduced is a hybrid integration methodology using PLM and P2P as a subset of the product development integration architecture. The remainder of this chapter therefore, will focus on the review of the technologies and methods used in the research.

2.3 ENTERPRISE SYSTEMS

This section describes the background of PDM/PLM and ERP systems, in particular an introduction of how the systems evolved from the traditional computer aided design and materials resource planning tools into the current enterprise-wide integration technologies.

2.3.1 CAD Tools to PDM

The reduction of product development cycle time and the improvement of design quality has been supported in the last two decades through the implementation of various computer aided technologies like 2-Dimensional and 3-Dimensional CAD (Computer Aided Design), CAE (Computer Aided Engineering) and CAM (Computer Aided Manufacturing). All these tools aimed to accelerate, automate and integrate various engineering and manufacturing processes. The introduction of CAD tools, however, was often only task instead of process oriented. Organisations tried to automate their tasks as efficiently as possible to create an environment which best suits and meet their targets. In practice, only little integration took place. Each CAD tool produced its own data format, and this problem could only partially be resolved by various interfaces and data exchange formats like IGES (Initial Graphics Exchange Specification) and STEP (STandard for the Exchange of Product model data) that were developed to facilitate communication between the different CAD tools.

Therefore, the result was often a heterogeneous, fragmented, multi-system environment in many companies and especially in design chains for product definition. Browne *et al.* (1988) state that "during the engineering supply chain, manufacturer and suppliers need to work as design partners, however their design and manufacturing systems are frequently incompatible. This introduces very significant time and cost penalties as they attempt to share design information." Moreover, the large amount of data generated by the various CAD tools was often not centrally organised and therefore inaccessible to others. As a consequence, additional costs for new product developments were inevitable without the usage of existing designed parts and data which can reduce the development time.

Realising this problem and the associated business opportunities, several software companies started in the mid 1980s to develop Product Data Management (PDM) systems (Philpotts 1996) that initially provided vaulting and file management capabilities for engineering documents like 2D CAD drawings. Prior to PDM, it was also known under several names such as Technical Information Management (TIM), Engineering Document Management (EDM), Product Information Management (PIM) and Engineering Data Management (EDM) (Philpotts 1996).

In the late 1980s and early 1990s engineering change management to control and track the changes made to engineering data was added to the functionality together with configuration and classification management capabilities. As a result of 3D CAD systems and teamwork approaches entering the market PDM systems started to support the management of complex relationships between parts, assemblies, drawings, metadata, people and groups of people. In the mid 1990s many PDM vendors switched their operating system from UNIX to Windows based and moreover various improvements in the user interfaces were made. This was followed by the additional capability of workflow management. Despite all the different terms used to represent PDM, they all share a common definition which "concerns the management of data across the product life cycle of a product; that is from its inception, development, test, and manufacture, through its ultimate demise, and ensure product data accessible to the right parties at the right time in order to support all business processes", (McIntosh 1995; CIMdata 1997; Burdick 1998; Goossenaerts and Pels 1998; Sackett *et al.* 1998).

2.3.2 PDM to PLM

In the late 1990's, there was a shift of focus in PDM technology, rather than on individual companies, the capabilities of PDM was enhanced into the supply chain areas, and was known as Collaborative Product Data Management (cPDM) and Collaborative Product Commerce (CPC) (Burdick, 1998), although, PDM was still the common name being used in industry and academia. Later, a new generation of software, Product Lifecycle Management (PLM) system has evolved from the PDM system.

A PLM system can be described as an enterprise-wide Information Technology (IT), "an infrastructure to support management of product definition throughout its complete lifecycle from initial concept to product obsolescence" (Portella 2000). Including workflow management, PLM systems, as a single source of product information, ensure that up-to-date information are available and accessible for the right people in the right format at the right time. It is also viewed as "an effective tool in managing the product definition supply chain by serving as an informational bridge connecting OEMs, partners, subcontractors, vendors, consultants and customers" (Miller 2003). To cope with the industry as well as social trends described above and to gain competitive advantage a new integrated approach for developing products with respect to the whole product lifecycle has to be taken. Integration has to take place internally (between different departments) as well as externally (i.e. with suppliers and customers). With the advances in user interfaces and databases, viewing as well as the Internet technology the technological prerequisites were provided to share data more easily. Providing integrated visualisation and Digital Mock-Up (DMU) Tools, PLM systems make it possible to view and mark-up native CAD data without the need for having access to the native CAD system. The Internet serves as a highly effective platform to communicate product data information far beyond the engineering organisation.

The introduction of a PLM system by itself, however, hardly initiates the changes that are required to improve the product development process. It is therefore important to recognise that "PLM is not a tool to improve the product development process but a tool to support an improved product development process. Before it can work to its full potential there must be changes in organisational processes as well as in the organisational culture (Stark 2004)."

The importance of the product development process, especially for manufacturing companies has been widely discussed (Smith and Reinertsen 1995; Paashuis 1997; Pawar and Sharifi 2000). PLM, especially CPC systems, as relatively new IT infrastructure to support an improved product development process are currently intensively discussed, especially among Research Firms (like the Aberdeen Group, AMR Research, Forrester Research and Burdick 1998) as an opportunity for manufacturing companies for major improvement and competitive advantage.

2.3.3 Issues of Implementing PLM Solutions in Collaboration and Sharing Organisational Knowledge

The organisational and cultural implications of implementing inter-organisational integration and collaboration via a PLM solution in sharing knowledge and data is emphasised by some researchers in this field. For example, Garetti *et al.* (2005) state that "There are a number of organisational and cultural obstacles to overcome to implement collaboration and sharing knowledge and information in PLM, one typical issue is people often resist new procedures". Similarly, Miller (2003) states that, "sharing data, knowledge or handing off ownership of data to others as the project moves through various phases of the product development is a potential cause for conflicts. Moreover, departments or groups that previously operated autonomously may resent having to co-ordinate their activities with others and that work styles and customs may vary between companies located in different areas around the world". Bourke (2001) in the same context emphasises, that collaborative product design especially implies the need for a high degree of openness and trust for best results, addressing cultural problems may be as important, or more important, than technical solutions.

2.3.4 The Journey of MRP to ERP

There were claims in the 1980s (Porter 1985) and early 1990s (Earl 1990) that Information Communication Technology (ICT) would change the way people and organisations conduct business. This has been proven to be the case as Jarvinen (1991) made the introduction of several information systems possible and necessary for doing business. In the history of the evolution and development of ERP systems, *Material Requirements Planning* (MRP) systems grew to *Manufacturing Resource Planning* (MRPII) systems (Chung and Snyder 1999, 2000) and these systems later evolved to Enterprise Resource Planning (ERP) systems.

MRP was developed in the early 1960s as the computerised approach to the planning of materials acquisition and production (Plossl 1995), for determining and planning material requirements for production, calculating and monitoring the materials and the time required to translate sales orders into finished products, detailing when individual items needed to be purchased. Later, MRP systems expanded into two further categories of supporting financial function which included transaction processing, purchase and inventory. A second function was to support operational decision making which included rough-cut capacity planning (RCCP), capacity requirements planning (CRP) and shop floor control. Further evolvement of MRP through priority planning to closed-loop MRP as new tools, including production planning, master scheduling, capacity requirements planning and the ability to execute materials and capacity plans were developed and integrated. This formed the basis for the next evolution to manufacturing resource planning (MRPII) in the 1980s (Kessler 1991) allowing manufacturing companies to optimise materials, procurement and manufacturing processes, and providing financial planning reports (Richardson 1988).

Significant productivity gains were possible by managing all production resources and not just materials (Goddard 1985). Leading edge technologies were integrated, including barcodes and radio frequency (RF) terminals to improve the capture, quality, accuracy and feedback of information and links with external databases and CAD were mooted to improve responsiveness of MRPII system models amid criticisms of inflexibility (Krepchin 1986). Key drivers for effective use in maintaining competitiveness in global markets were developed, including quality management, high data integrity, planning and control, flexible production and teamwork (Luscombe 1993), driven by philosophies such as total quality management (TQM) and concurrent engineering (CE). These formed the basis for the next evolution in manufacturing planning systems known as nowadays, Enterprise Resource Planning (ERP).

ERP systems are highly integrated software packages (Holland *et al.* 1999) that can be customised to cater for the specific needs of an organisation (Boudreau and Robey 2000; Esteves and Pastor 2001). Additional tools and applications to encompass the whole of the business operations of an organisation, including reporting and analysis, human resources, forecasting, quality control, customer relations management (CRM), advanced planning and scheduling (APS), monitoring and finance became integrated (Puchalski 1997). The ERP paradigm signifies the convergence of technology and system architecture (client/server) with the integration of software functionality for all departments and operations in the business enterprise, developed around growth technologies like the Internet to broaden the scope into the supply chain.

2.4 AGGREGATE PROCESS PLANNING CONCEPT

Aggregate Process Planning is a methodology, developed at the University of Durham, for the selection of the most appropriate processes and resources and the automatic creation of "rough-cut" processing information from early feature-based product models (Maropoulos *et al.* 1998; Yao *et al.* 1998). The purpose of Aggregate Process Planning is to allow alternative process plans (or routings) for custom parts to be generated, evaluated and improved based on estimated *manufacturability* before committing to a fully-specified product model and supplier.

Yao *et al.* (1998) and Maropoulos *et al.* (1998); (2000a and b) developed the CAPABLE aggregate process planning tool-kit for integrated product development and CAPABLE/Welding that can be used to evaluate the design of sheet metal fabrication at the early stages. Bramall *et al.* (2001a and b) developed a Knowledge-Enriched CAPABLE Aggregate Process Planning system for the aerospace applications by capture and representation of product and process knowledge during aggregate planning and subsequently prioritising knowledge using the theory of Capability Analysis (Baker and Maropoulos 1998; Bramall *et al.* 2002) to produce a realistic process plan as shown in Figure 2.1.

CAPABLE/space is a software implementation of Knowledge-Enriched Aggregate Process Planning which utilises intelligent planning algorithms to automatically explore process and resource alternatives from the enterprise model, seeking a process plan which results in near-optimal manufacturability and interaction as determined by quality, cost, delivery and knowledge (QCD+K) criteria. Figure 2.1 presents the software modules and interaction required to implement the specified functionality, namely:

a) Modelling of products, process and resources for the enterprise.
b) Intelligently optimising the selection of processes and resources for a given product.
c) Representing the manufacturing implications of knowledge related to products, processes and resources.

d) Managing and prioritising knowledge to increase the effectiveness of the design process.

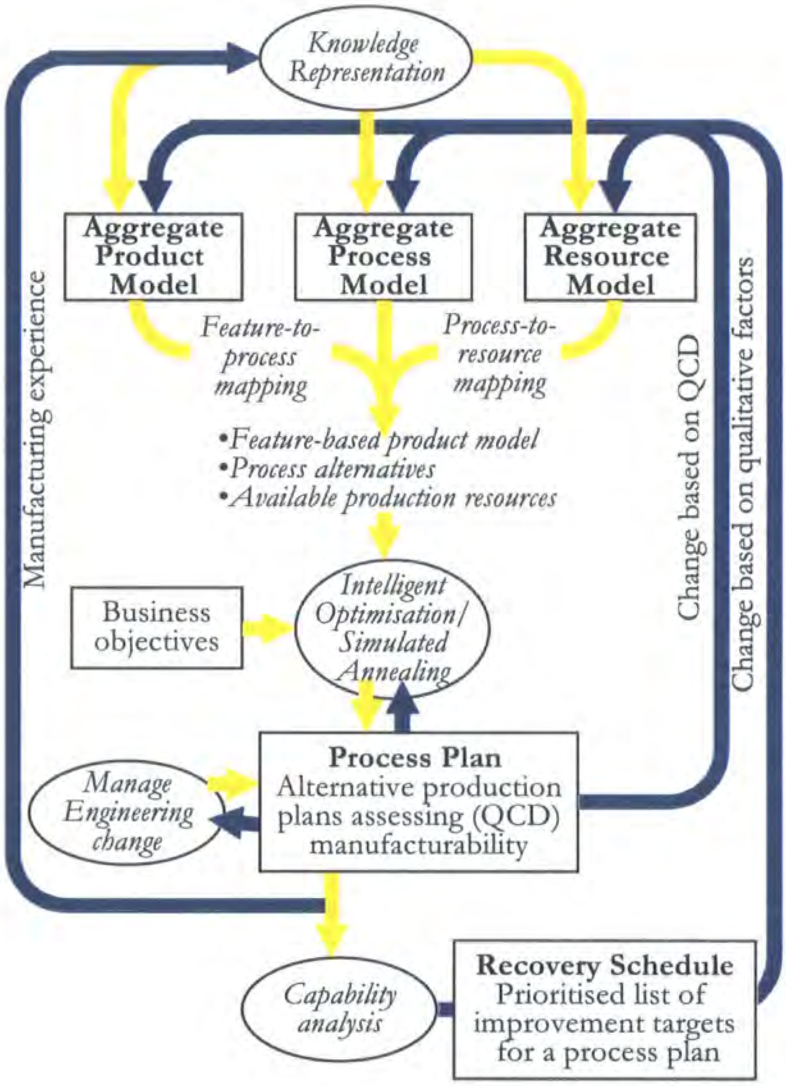


Figure 2.1: Information Flow through the CAPABLE/Space System. Reproduced from Bramall *et al.* (2001a)

2.4.1 The Aggregate Product Model

An Aggregate Product Model (Yao *et al.* 1998; McKay *et al.* 2000) is defined by a hierarchy of components and manufacturing features such as holes, pockets, slots etc which contain information about a design's fundamental geometry and feature relations between components. Making changes to the product model allows different product configurations to be evaluated. A component is made up of positive features, negatives, joint features and sub-components as shown in Figure 2.2. The aggregate product model is a tree of components and feature objects, which serves to represent various products. The aggregate product model allows the effective representation of early product configurations and manufacturing features including:

- Bill-of-Materials-like assembly structure.
- Key manufacturing features with critical geometries identified.
- Un-related tolerances on critical integration features.
- Positive features with manufacturing related materials data.

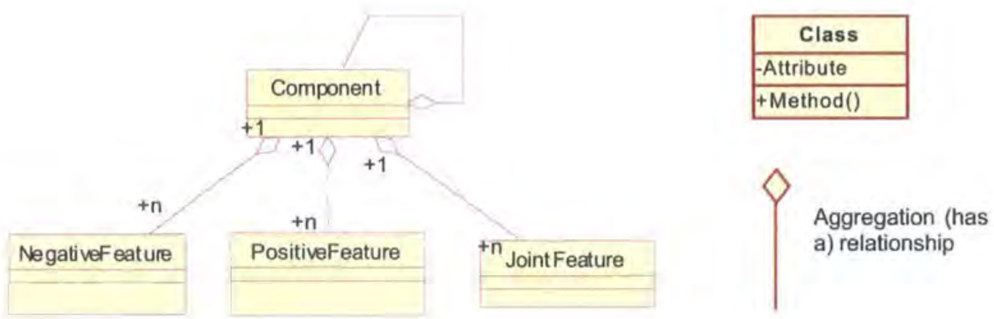


Figure 2.2: Representation of the UML Product Model Structure

2.4.2 The Aggregate Process Model

An Aggregate Process Model (Maropoulos *et al.* 2003a; Bramall *et al.* 2001a; McKay *et al.* 2001a) captures the fundamental aspects of process behaviour, complete with a selected set of process parameters and manufacturing knowledge. The aim of process modelling is the estimation of process performance in terms of Quality, Cost and Delivery (QCD). In aggregate process modelling the key production-related process variables are linked to characteristics of the product model. Heuristic rules for sequencing manufacturing operations also form a key part of process planning and are encapsulated within the process model.

2.4.3 The Aggregate Resource Model

An Aggregate Resource Model (Maropoulos *et al.* 2003b; Bramall *et al.* 2001b; McKay *et al.* 2001a and b) defines the equipment, cell and factory resources available for production. Essentially, a resource model is a database containing a description of the resource information necessary to support the process model. The type of information required includes: operating parameters, statistical process control data and financial information.

2.4.4 The Aggregate Assembly Process Planning

Betteridge (2000) developed an Aggregate Process Planning for assembly. This research is to derive a methodology to aid the generation of optimal assembly planning process during the early design stage for a given product. Laguda (2002) further developed this work by identifying the best possible approach to developing and building a product or part, with the ultimate aim of reducing assembly times and assembly cost of the part based on available resources, the system is named CAPABLE*Assembly*.

2.5 KNOWLEDGE MANAGEMENT

The arrival of the information society and the move towards the knowledge-based economy highlighted the importance of knowledge and the needs to manage knowledge resources including skills and competencies. Knowledge Management (KM) brings a new dimension by focusing on people and enhances their capability by improving communication, information transfer and collaboration (Al-Hawamdeh 2002). The term KM is well documented in the business literature initiatives (Davenport and Prusak, 1998). What is actually entailed in these initiatives remains ambiguous because there are many interpretations of knowledge management. A review by Hlupic *et al.* (2002) has identified 18 different definitions of KM. Many attempts have been made to define KM from a theoretical perspective (Choo 2006; Srikantaiah and Koenig 1999; Oluic-Vukovic 2001) and to identify the various types of organizational knowledge (Nonaka and Takeuchi 1995; Brown and Duguid 1998).

2.5.1 Knowledge and Knowledge Management

Here are a few of the definitions of knowledge and KM. According to Davenport and Prusak (1998), knowledge is a fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. In organisations, it often becomes embedded not only in documents or repositories but also in organisational routines, processes and practices.

Schmidt (2000) defined KM is an integrated, systematic approach to identifying, managing and sharing an enterprise's information assets, including documents, databases, policies, procedures, and implicit expertise. The elements of this systematic approach are illustrated in Figure 2.3:

1. Problem formulation: define the scope of the knowledge effort.
2. Search: locate currently available information and expertise.
3. Organize: analyze and categorize the information.
4. Create: create new knowledge and understanding.
5. Capture: make the knowledge explicit and make it available for others.
6. Reuse: update the knowledge structure so others can reuse this knowledge (now converted into information).

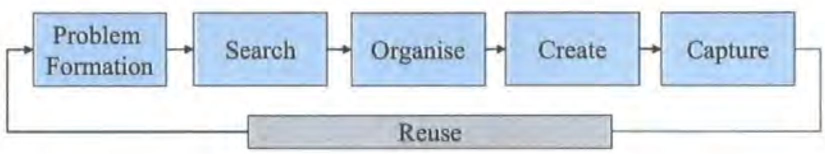


Figure 2.3: Knowledge Management Systemic Approach

Dvir and Evans (1998) state that reuse of knowledge is "the application of existing solutions to new problems". Reuse can directly contribute to three critical success factors of the product development process: Time to Market, Cost (both unit cost and development cost), and product Quality. They also suggested that reuse must be managed in a systematic way. Similarly to the way quality is managed i.e. they propose to apply well established quality concepts and methods to the measurement of reuse. A "Reuse Atlas" that provides both *quantitative metrics* and *qualitative information* on reuse is outlined as a major tool in a "Total Reuse Management" framework.

Eyler (2001), states that KM is a discipline that promotes an integrated approach to the creation, capture, organization, access, and use of an enterprise's information assets. These assets include structured databases, textual information such as policy and procedure documents, and most importantly, the tacit knowledge and expertise resident in the heads of individual employees. Authors such as Gourlay (2000) and Beckman (1999) present KM as an emerging discipline. Gourlay (2000) suggests that KM practices focus mainly on knowledge representations and, according to Beckman, the expression was coined by Karl Wiig who wrote one of the first books on the topic, *Knowledge Management Foundations*, published in 1993.

On the other hand, others, such as Streatfield and Wilson (1999), claim that firms and information professionals have been practicing KM-related activities for years. Streatfield and Wilson (1999) argue that the concept of knowledge is over-simplified in the KM literature, and they seriously question the attempt to manage what people have in their minds. Nevertheless, there is a real interest and enthusiasm in KM as revealed by the increasing number of publications relating to the topic since 1995 (Mahdjoubi and Harmon 2001), a typical review was by Wilson (2002), Figure 2.4.

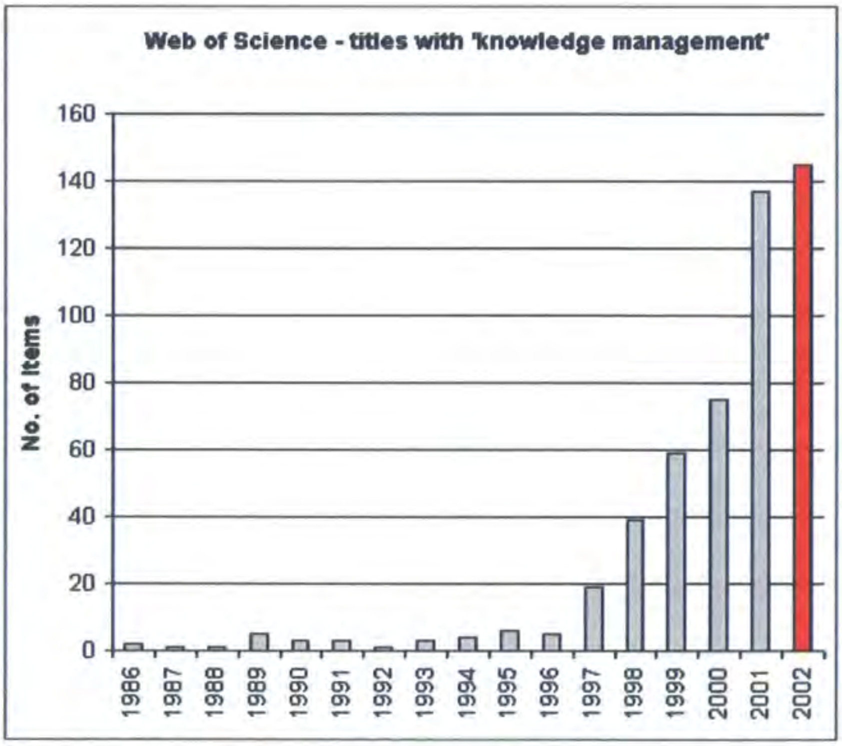


Figure 2.4: Emerging Publications in Knowledge Management (courtesy of Wilson, 2002)

2.5.2 Research Towards Organisational Knowledge and Management

Part of the research interest within this work is the application of KM to capture organisational knowledge. Organizational knowledge is frequently categorized into typologies, for example, Nonaka and Takeuchi (1995) identify tacit and explicit knowledge; Choo (2006) sees three different types of knowledge (tacit, explicit, and cultural); Tacit knowledge is defined as action-based, entrenched in practice, and therefore cannot be easily explained or described, but is considered to be the fundamental type of knowledge on which *organisational knowledge* is built (Nonaka and Takeuchi, 1995; Choo 2006). Explicit knowledge, unlike tacit knowledge, is defined as knowledge that can be codified and therefore more easily communicated and shared. On the other hand, Choo states that "Cultural knowledge is defined as the shared assumptions and beliefs about an organisation's goals, capabilities, customers and competitors. These beliefs are used to assign value and significance to new information". KM writers view explicit knowledge as structured and conscious and therefore it can be stored in information technology (Martensson 2000).

By definition (Wikipedia 2006) "Tacit knowledge is knowledge that people carry in their minds and is, therefore, difficult to access. Often, people are not aware of the knowledge they possess or how it can be valuable to others. Tacit knowledge is considered more valuable because it provides context for people, places, ideas, and experiences." Organisational knowledge is being adopted as part of the work in this research in particularly modelling tacit knowledge or *'know-how'*. According to ITAG 1999, *'know-how'* which refers to skills and capabilities is, for example, the ability to use a particular machine or skills gained through practice and experience. In industry, '*know-how*' is often used interchangeably with the term knowledge management from design to manufacturing, whether its context is described explicitly or implicitly.

2.5.3 Converting Tacit and Explicit Knowledge

Attempts to define KM processes are numerous. Nonaka and Takeuchi (1995) described four knowledge conversion processes: socialization, externalization, combination, and internalization. Each process involves converting one form of knowledge (tacit or explicit) to another form of knowledge (tacit or explicit). This model focuses on the important issue of how knowledge may be created through organisational sharing and is useful for identifying and evaluating certain key activities in the management of knowledge. Hlupic et. al. (2002) identified three main types of activities: knowledge generation involving the creation of new ideas and new patterns; knowledge codification, and knowledge transfer, ensuring exchange of knowledge between individuals and departments.

Another model, proposed by Oluic-Vukovic (2001) outlines 5 steps in the knowledge processing chain: gathering; organising; refining; representing; and disseminating. This model covers more completely the range of activities involved in the organizational knowledge flow. A KM conceptual framework developed by Bouthillier and Shearer (2002) has particularly addressed the issues of the processes of knowledge acquisition to be reused and shared is shown in Figure 2.5. The purpose of the study was to identify general trends in KM practices across several organisational types in order to gain insight into why and how organisations are practicing the management of knowledge.

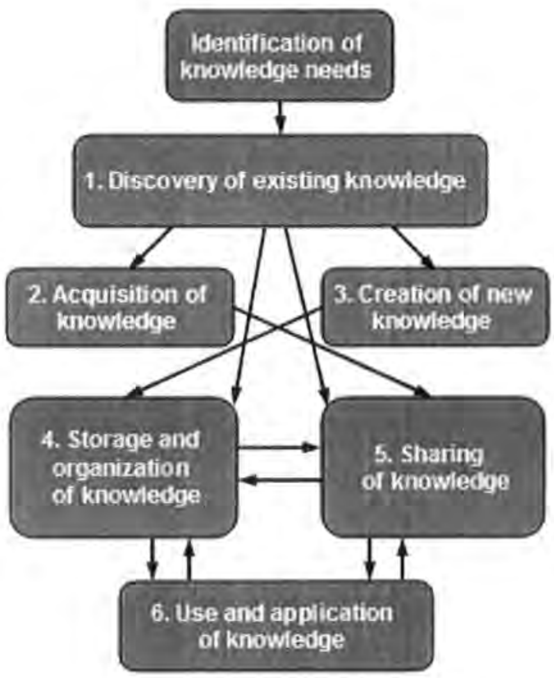


Figure 2.5: Conceptual Framework: Knowledge Management Processes (courtesy of Bouthillier and Shearer 2002)

Apart from Bouthillier and Shearer (2002), none of these models are broad enough to allow for a complete analysis of organizational knowledge flow, omitting several important steps in the knowledge chain, such as acquiring and storing knowledge. One of the main objectives of the work in this research is to be able to store and reuse manufacturing knowledge to support the product development processes. Thus, the understanding and background of the above KM techniques and reviews provide the foundation to achieve this.

2.6 INTRODUCTION OF ONTOLOGICAL APPROACH IN KNOWLEDGE MANAGEMENT

There has been significant amount of research in the field of knowledge capture, sharing and re-use to support the product development process within a global business enterprise. The techniques developed for these can be categorised into; rule-based systems, case-based systems, model-based reasoning, neural-nets, fuzzy logic, decision trees (Turban and Aronson 2000) and ontologies (Hausser 2000). Ontologies are increasingly becoming important in the fields of intelligent searching on the web, knowledge sharing and reuse, and knowledge management. In information technology, an ontology is the working model of entities and interactions in some particular domain of knowledge or practices, such as electronic commerce or the activity of planning (Davies *et al.* 2002). In Artificial Intelligence (AI), according to researchers at Stanford University, ontologies can be used to express "a set of concepts such as things, events and relations that are specified in some way in order to create an agreed vocabulary for exchanging information, in particular over the World-Wide-Web (WWW)" (http://www.w3schools.com). Apart from providing a common understanding, Valarakos *et al.* (2004) also state that ontologies can be used to facilitate dissemination and reuse of information and knowledge. The main technologies used to derive an ontology are the Process Specification Language (PSL) (Schlenoff *et al.* 2000) and Web-based technologies. The standards of Web-based technologies are eXtensible Markup Language (XML), Resource Description Framework (RDF), Web Ontology Language (OWL) (http://www.w3.org) and XML Metadata Interchange Format (XMI) (http://www.omg.org).

Ontologies have been used to share and reuse knowledge and information, predominately in the medical informatics field. The main reason ontologies have become so popular is the fact that they provide a shared and common understanding of a domain that can be communicated between people and application systems (Davies *et al.* 2002). Lately, there have been an increasing number of research projects applying ontological techniques in the context of product development (Moore *et al.* 1999; Duineveld *et al.* 2000; Roche 2000; Ciocoiu *et al.* 2001 and Lin and Harding 2003).

2.6.1 Ontology-Based Applications

There is an increasing volume of research in ontology-based applications of knowledge management and information sharing and retrieval. The list of applications presented here is not exhaustive, but, it highlights some of the more interesting work from different areas of the ontology based research and development community.

An Intelligent Brokering Service for Knowledge-Component Reuse on the WWW (IBROW3 2004) is an ontological-based knowledge system. The objective of IBROW3 is to develop intelligent brokers that are able to configure reusable components into workable knowledge systems through the WWW. In the knowledge engineering discipline, the On-To-Knowledge (OTK 2004) project has developed practical methods and tools, based on an ontological approach, to facilitate knowledge management as a means to share and reuse knowledge. The OTK tools help knowledge workers, who are not IT specialists, to access company-wide information repositories in an efficient, natural and intuitive way. The OTK project applies ontologies to electronically available information to improve the quality of knowledge management in large and distributed organisations.

"Production d'Interfaces à base de Connaissances pour des Services En Ligne" (PICSEL) (Goasdoue 1999) is an information integration system for knowledge sources that are distributed and possibly heterogeneous. The approach taken was to define an information server as a knowledge-based mediator between users and existing information sources relative to a single application domain. OntoBroker (2004) is a reasoning engine with semantic information integration capabilities from Ontoprise GmbH (www.ontoprise.com, Germany). Data integration is done via several connectors, import and export formats and built-in functions, for example, importing data schemas from existing databases, mapping to ontologies and connecting to search engines and applications. SemanticMiner (Moench 2003; Moench *et al.* 2003; SemanticMiner 2004) is a knowledge retrieval platform that combines semantic technologies with conventional retrieval approaches. It is designed as client-server architecture and provides information retrieval from various data sources. The SemanticMiner-Server, which is a specialized OntoBroker-system, provides the interface to the data sources as well as an inference engine to retrieve implicit knowledge.

The above research provided the foundations of the work to be described in Chapter 6, in particular, the application of the ontological approach to capture manufacturing knowledge, share and reuse it in the WWW (World Wide Web) which has been developed in a manufacturing context. This research was focused on creating a single knowledge domain for manufacturing similar to *Goasdoue's* approach. The method of sharing knowledge in this work, however, is to make use of existing technology such as Product Lifecycle Management which allows the same information to be shared heterogeneously within a collaborative and distributed secure environment. Another aspect of this work described is the adoption of the approach similar to *Moench's* work which enables users to retrieve knowledge for various applications. The method of knowledge retrieval however, is to make use the application of XML Parser technology which enables the XML-formatted knowledge to be shared by other third party applications. Therefore, the main objective of this part of the research was to exploit the benefits of a single manufacturing knowledge repository with a standard data format for use by multiple, distributed collaborative systems to support the product development process.

2.7 APPLICATION OF WEB-BASED TECHNOLOGIES

2.7.1 RDF and XML

Resource Description Framework (RDF) (W3C, 1999) is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web. RDF emphasizes facilities to enable automated processing of Web resources. RDF can be used in a variety of application areas; for example: in *resource discovery* to provide better search engine capabilities, in *cataloguing* for describing the content and content relationships available at a particular Website, page, or digital library, by *intelligent software agents* to facilitate knowledge sharing and exchange. RDF with *digital signatures* will be key to building the "Web of Trust" for electronic commerce, collaboration, and other applications.

One of the goals of RDF is to make it possible to specify semantics for data based on XML in a standardised, interoperable manner. RDF and XML are complementary: RDF is a model of metadata and only addresses by reference many of the encoding issues that transportation and file storage require (such as internationalisation, character sets, etc.). For these issues, RDF relies on the support of XML. It is also important to acknowledge that, currently, XML syntax is the only possible syntax to support and represent RDF.

As a result of many communities coming together and agreeing on basic principles of metadata representation and transport, RDF has drawn influence from several different sources. The main influences have come from the *Web standardisation community* itself in the form of HTML metadata and Platform for Internet Content Selection (PICS), the *library community*, the *structured document community* in the form of Standard Generalized Markup Language (SGML) and more importantly XML, and also the *knowledge representation (KR) community*.

There are also other areas of technology that contributed to the RDF design; these include object-oriented programming and modelling languages, as well as databases. While RDF draws from the KR community, readers familiar with that field are cautioned that RDF does not specify a mechanism for *reasoning*. RDF can be characterized as a simple frame system. A reasoning mechanism could be built on top of this frame system.

2.8 INTEGRATION OF APPLICATIONS

2.8.1 Introduction of Application Integration

Integration of applications, information, and business processes have become today's first priority investment of Information and Communication Technologies (ICT) (Linthicum 2003). "As enterprises shift their focus beyond mere application integration to getting processes to work and play well together, they need specialists who can see the big picture to assure they get things right", says IBM's Douglas W. Allen. However, Linthicum has identified that over 90 per cent of total software lifecycle cost is in maintenance and integration. Despite the cost, integration is an important issue of aligning business requirements to reduce risk and maximize return on investment (ROI) (Gold-Bernstein and Ruh 2004).

2.8.2 Integration Paradigms

Due to rapid advances in ICT, new paradigms have been proposed to support distributed production network environments. Among these are: Virtual Enterprise (Ettighoffer 1992) Distributed and Re-configurable Enterprise (Gunasekaran 1998) and, most recently, Cloutier *et al* (2001) introduced Networked Manufacturing Co-ordination and Vernadat, (2002) on Enterprise Integration architecture. All these paradigms have one thing in common; they all require highly trained and sophisticated skills to customize and integrate with other systems. Customization is often required to modify the software to suit the environmental needs. This may cause the system to be more unstable and difficult to maintain when it finally comes online. In addition, organizations need to become expert in the art of integration in order to meet their business needs and compete effectively.

With the maturity of Web-based technologies and the availability of 'off-the-shelf' enterprise systems, this research has proposed a different approach: to deploy an 'out-of-box' solution for the integration of disparate software systems. The term 'out-of-box' means a solution is ready to be implemented without the needs of further customization of the software. The advantages of this approach are rapidly reconfigurable collaboration networks and minimal requirements for system customization, thus avoiding instability that might result.

2.9 APPLICATIONS OF WORKFLOW TECHNOLOGY

2.9.1 Background and Applications of Workflows

Workflow management theory dates back to the late 1980s when diagrammatic process representation began to be used in a few pioneering organisations to support basic business processes (Beizer 2002). Plesums (2002) states that "workflow is the automation of internal business processes or operations, in whole or part, during which documents, information or tasks are passed from one participant to another for action and tracking its progress according to a set of procedural rules."

Extensive literature reviews have shown that the majority of applications utilising workflow analysis techniques are mainly centred around providing computer support for an organisation's business processes (Sayal *et al.* 2001; Moore and Graham 2002). Such business processes include customer order processing, product support in marketing, stock taking and processing. However, little work has been done in applying the techniques enterprise wide in particular to support the design and manufacture and product development process which would benefit from these tools. Although research has been carried out on cooperating workflow and Web-based technologies, the scope of the applications are narrowly focused and limited to business processes. For instance, Abecker *et al.* (2000) demonstrated two approaches to utilizing Web-based technologies and workflow techniques. The VirtualOffice scenario emphasized the use of workflow context information for improved document analysis and information extraction. The KnowMore approach supports a person working on some knowledge-intensive task by actively delivering context-sensitive and relevant information. Sayal, *et al.* (2001), describe how workflow technology and Business-to-Business (B2B) Standards such as RosettaNet, Common Business Library (CBL), Electronic Data Interchange (EDI), Open Buying on the Internet (OBI), and commerce eXtensible Markup Language (cXML) can be extended in order to support B2B interactions and to link them with the internal workflows. Cheung *et al.* (2003) examined the requirements of extending a workflow to the provision of e-services to fulfil predefined business processes and data requirements. The work proposed by Liu and Shen (2003) is based on a process-view coordination model that can be extended to effectively address the issues of managing B2B workflows.

2.9.2 Workflow Methodology

Workflow is a methodology for modelling analysis, assessment and reorganisation of working processes taking place within an organisation which involves the coordination and collaboration of different individuals as represented in Figure 2.6. The basic idea of Workflow is to describe the work and the activities taking place in an organisation as a business process, namely a "structured measured set of activities designed to produce a specified output for a particular customer or market" (Davenport 1993).

Another aspect of workflow is the improvement of working processes and its coordination through the introduction of Information and Communication Technology (ICT) and cooperation amongst individuals involved in the work processes.

Figure 2.6: Workflow System Characteristics

2.9.3 Related Approaches in Utilising Workflows

This section describes different approaches to workflow which have been proposed and implemented in different areas other than business processes. Malamateniou and Vassilacopoulos (2003) present a Virtual patient record (VPR) framework that allows the integration of geographically dispersed medical information within a health district and enhances collaboration and coordination of authorised workgroups by means of a Web-based workflow system. VPR provides a means for integrated access to patient information that may be scattered around different healthcare settings. In this context, a VPR implementation enables autonomous healthcare providers to operate in a cooperative working environment and apply continuity of care. Workflow systems bring this collaboration and co-operation into effect by automatically routing the medical information needed by authorised actors in a healthcare process.

Tunnicliffe and Maropoulos (2003) implemented a Cost Data Warehouse (CDW) strategy to share global cost reduction data for an automotive manufacturer. CDW allows cost reduction ideas to be effectively communicated and managed across departments such as design, purchasing and parts suppliers on a European and global scale to continually reduce the cost of manufacture throughout the vehicle life cycle. The system utilizes Web-based technologies, business process re-engineering techniques and links to a mainframe system, which contains product design data of live part numbers to allow users to deal with the issues in real-time. Huang *et al.* (2000) proposed to use workflow management as a mechanism to facilitate teamwork in a collaborative product development environment where Web-based Decision Support Systems are extensively used by geographically distributed team members. The concept used is workflow management supported by agent technology to prevent inconsistency and reduce redundant project activities. It can improve coordination and interaction of the engineers who work remotely in different locations.

Chung *et al.* (2003) investigated the use of ontologies, agents and knowledge-based planning techniques to provide support for adaptive workflow and flexible workflow management, especially in the area of new product development within the chemical industries. The work in this project uses a Task Based Process Management (TBPM) workflow engine, the Task Manager, to support heterogeneous agents (humans and software agents) in working together to achieve their tasks. The Task Manager includes a process modelling tool and an interactive process planner. The planner uses artificial intelligence techniques to assist in the planning of tasks, while permitting the user to participate in planning decisions. An agent-based architecture supports the execution and coordination of the planning process among multiple participants distributed across a computer network. TBPM also provides a process management framework that integrates with specialist third-party software systems such as simulation and analysis tools which are essential in chemical engineering.

None of the systems described, however, are currently exploited in the area of utilizing advanced tools such as a web-centric PDM system, which provides an easier and faster implementation of workflow modelling. PDM systems offer a Workflow function which enables the user to create a pattern of 'automation' to assign alternative tasks and eventful sequences as dictated by a company's operational procedures. Most of all, the workflow function can be used to define and implement time-dependencies of activities within a distributed team to support collaborative product development.

2.10 PEER-TO-PEER TECHNOLOGY IN COLLABORATIVE PRODUCT DEVELOPMENT

2.10.1 Principle of P2P

P2P is not a new concept in the networking world, but its application to the internet is a fairly recent development. P2P applications address the needs of de-centralised organisations to collaborate and share knowledge and files regardless of geographical location (Bond 2001). The principle of P2P has been around for a long time, and is today implemented in a number of applications such as instant messaging and file-sharing (www.GNUtela.com).

There are already a number of P2P PDM in existence, primarily aimed at the lower end of the market. The two applications are AutoManager workflow from Cyco (www.cyco.com) and Columbus from Oasys Software (http://www.oasys-software.com/). The latter is available for free, and aimed at AutoCAD users within the construction sector. There is no workflow or process management implemented and, access control is via the standard operating system access control functions. As an example of what can be achieved, Alibre is a P2P CAD/PDM and collaboration tool in one. It uses the STEP standard and combines low cost and fast configuration.

The advantages offered by P2P applications are (Bond 2001; Penserin *et al.* 2003):

- No single point of failure, the network is alive as long as one peer is on-line
- Distributed sharing of bandwidth, storage and processing power, so the system becomes more powerful as more users attach
- Lower running cost due to the lack of servers or high bandwidth central nodes, as well as
- Maintaining individual control of the shared knowledge.

P2P groups can be used to create profiles of the peer. These profiles can be used within the network to search for and assess people's competences, interests, and memberships of trusted groups, and can aid in the construction of new relationships based on commonalities and third party assessments. There have been a number of issues that reduce the performance of the system when using a pure P2P architecture. The lack of indexing and routing services in P2P degrades the peer discovery and query functions (Penserin *et al.* 2003). In order to leverage the advantages of client/server systems with the independence and interoperability of P2P systems a hybrid system where "super peers" act as peers to the extended P2P network and as a server to the enterprise's internal peer network is used. In addition, rendezvous peers can be assigned to manage some of the peer information assigned to particular peer nets or projects. These hybrid systems have the best potential for high-performance de-centralised services.

2.10.2 Application of P2P Technology

The application of P2P technology in industry is still in its infancy. However, there is an increasing volume of research in P2P-based applications to address information sharing in different industries. The list of applications presented here highlights some of the more interesting work from different areas of the P2P based research and development community.

SwinDew, (Yan *et al.* 2003) combined the concepts of workflow and P2P to present an innovative decentralised architecture. The approach enables team members to be supported with coordination by direct communication among peers, which is claimed, to enhance the effectiveness of distributed applications. SWAP, (Ehrig *et al.* 2003) is another project which combines two highly successful technologies; semantic web and P2P computing to address sharing knowledge between organisations in a decentralised network. The approach allows each company to maintain their knowledge structure while exchanging information, which cannot be achieved through a centralised network. The SATINE project (Satine Consortium, 2003), provides a component to wrap existing information resources to make them appear as semantically well described Web Services. SATINE provides an easy to use tool for Small and Medium-Sized Enterprises in tourism industry to easily create Web Services from their existing enterprise applications. The wrapped resources will be able to exchange information with other Web Services on a Peer-to-Peer node (Bond, 2001). However, all the projects do not directly address the issues related to the collaborations of SMEs and OEMs in product development, in particular there is no research activity targeted at finding new ways for individuals to communicate with others via the Internet using the P2P approaches which would have a significant impact in the product development processes.

2.11 SUMMARY

Setting the scene for a chapter that reviews important concepts and technologies in collaborative manufacturing, requires a variety of topics to be introduced. The selection of these topics has been quite diverse and complicated due to the broad area of research. This chapter has presented the background and the importance of key elements to support an enterprise-wide approach in collaborative product developing environments. Among the research reported in Section 2.1, Huang *et al.* (2000), (2001), (2002) and Li *et al.* (2004a), (2004b) are the main authors who are focused on research in collaborative product development.

The existing system CAPABLE, shows a great deal of potential to examine preliminary process planning configurations in early product design and development stages. The applications of knowledge management in organisations have highlighted the importance of knowledge and the need to manage knowledge resources including skills and competencies. However, the lack of trust and cultural implications to sharing knowledge are still the main issues facing industry to apply KM. The approaches of using ontologies in organisational knowledge are relatively new in product development and, it has indicated a great deal of benefits. A commercial software system using ontology techniques to manage organisational knowledge in product development is not available at present. However, the application of an 'out-of-box' solution for the integration of disparate technologies will be the key business benefit for rapidly reconfigurable network and minimal requirements for software customization to avoid systems instability.

The maturity of Web-based data formats is becoming more apparent, for example, XMI, RDF, and XML are the standards this research work has adapted. The application of enterprise systems is not just a system, it is a complete strategy to provide effective and reliable information flows to internal as well as external areas of the company. The introduction of such technology, nevertheless still needs the support and involvement of top management.

The increasing influence of networking environments and the growing concerns of smaller companies to collaborate with large companies have motivated the introduction of the Peer-to-Peer technology. The application of Peer-to-Peer technology is to demonstrate the advantages of using a de-centralised production network configuration.

Finally, this chapter has described the background of similar research and the key elements of the technologies. In the next section, Chapter 3 will look at the assessment of an industrial survey and the enterprise integration system which is the main component to create the product development integration architecture to address a distributed and collaborative product development.

Chapter 3

PRODUCT DATA MANAGEMENT SYSTEM AND INDUSTRIAL SURVEY

3.1 INTRODUCTION

This chapter examines the role of the PDM technology, in particular the impact of PDM on industry. In addition, this chapter will present the results of the industrial survey. The survey was designed to measure how industrial companies have implemented PDM and what they felt the benefits of implementing PDM technology were. The main application of the enterprise integration system in this research work is to form an integration architecture with other third-party software systems to evaluate product manufacturability and bridge the discontinuity of early design processes.

3.2 WHAT IS PLM/PDM?

PDM systems are about managing product information, throughout the entire lifecycle of a product, in a more efficient, organized way (Little and Lee 1999; CIMDATA 2001; Liu and Xu 2001). A PDM system is used to store and process product design data related to its products, and manage the life cycle of those products. It provides the mechanism to capture and enforce the specific product development process consistently according to the way in which a company does business. The goal is to provide a much richer feature set and knowledge base to an even larger community of users, whether they work in a manufacturer's engineering centre, at an overseas subsidiary, at a supplier or other business partner. The challenge is to maximize the time-to-market benefits of concurrent engineering while maintaining control of data and distributing it automatically to the people who need it, when they need it.

The theoretical evolution of PLM has been described in the literature review. PLM can be classified as the offspring of PDM technology. However, PLM technology has transformed into individual modules that serve the needs of different parts of the product development processes in its entire product life cycle, from conceptual design to manufacturing, customer delivery, maintenance and disposal. Burkett *et al.* (2002 and 2003) reported that PLM consists of different software modules to support five core areas, these are:

- Product Data Management (PDM)-Product information repository
- Collaborative Product Development (CPD)-Design of products and processes
- Direct Materials Sourcing (DMS)-Sourcing of standard or custom parts and materials
- Customer Needs Management (CNM)-Managing customer or market input
- Product Portfolio Management (PPM)-Managing resources in product development

A more comprehensive description of each of the core areas is shown in Appendix A.

3.2.1 The Benefits of Using PDM

PDM provides benefits across every area of product design and development. Organisations can gain an advantage through using this technology. PDM systems and methods provide a structure in which all the different types of information used to define, manufacture, and support products are stored, managed, and controlled. Typically, PDM is used to work with database records, electronic documents, and digital files (Philpotts, 1996). These may include:

(a) Database Records

- Product configurations
- Part definitions and other design data
- CAD drawings
- Geometric models
- Product Specifications

(b) Electronic Documents

- Engineering analysis models and results
- Manufacturing process plans and routings
- Images (scanned drawings, photographs, etc.)
- Software components of products
- NC part programs

(c) Digital Files

- Electronically stored documents, notes, and correspondence
- Hardcopy (paper-based and microfilm) documents (by reference)
- Project plans
- Audio and live video annotations

According to Little and Lee (1999), in the UK, PDM technology is used by 50% of the aerospace companies, 35% of the automotive industry and 15% of smaller high-tech, high growth manufacturers. The main usages of PDM in industry according to Hall (2000) are 89% for CAD management, 17% for documentation and spreadsheets and 49% in bill of materials (BoM), specification management and configuration management. Since these comparisons were obtained several years ago, the PDM technology has evolved to the current PLM solutions, and the technology covers an even wider area from the original product data solutions to customer relation and supply chain management. Thus, this indicates that the technology has improved and, is moving towards a global collaborative level - a technology that must be adapted if a company is seeking to enhance their competitiveness in the global market.

3.2.2 Commercial PDM Functionality Overview

Figure 3.1 illustrates the core functionalities of the PDM technology. PDM was first introduced in the early 1980's; the systems were developed based on a set of functions that included user, utility and electronic vault (Philpotts, 1996). Since the emergence of Internet technology in the mid 1990's, the concept of PDM technology has evolved and encompasses Web-based collaborative and e-Integration functionalities (CIMdata, 2001). Details of a more comprehensive description of each individual function are available in Appendix B. An evaluation of the functionalities of various commercial PDM systems is shown in Appendix C.

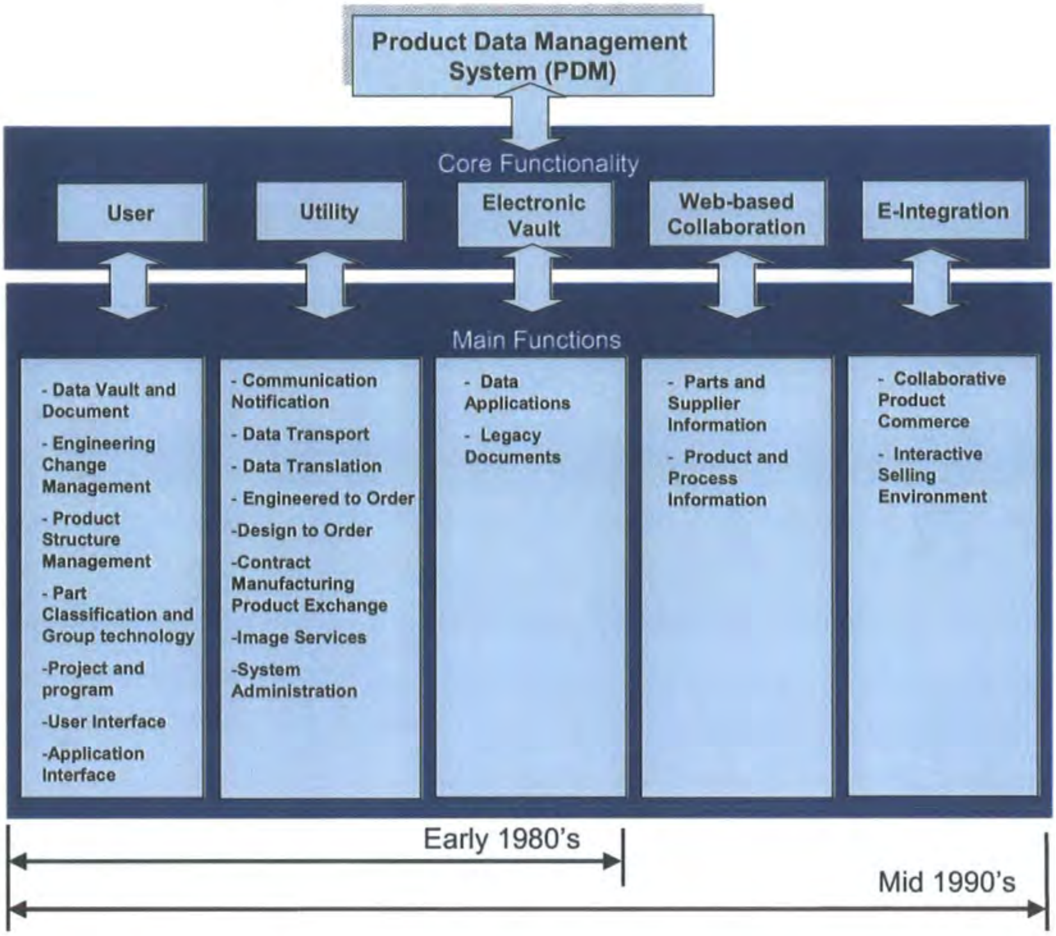


Figure 3.1: PDM Functionality Diagram

3.3 THE PDM INDUSTRIAL SURVEYS

To ascertain the current approaches used within industry a survey was undertaken. For the survey a number of approaches could be adopted such as semi-structured interviews or email/postal questionnaires (Denscombe 1998). To ensure that the survey was undertaken in a structured approach a review of 'best practice was undertaken'. In general the approach to adopt consisted of seven steps as illustrated by Oppenheim (1992) and Burgess (2001). This approach was used within this research to ensure that the most appropriate method for surveying was adopted and that the outcomes required from the survey were met. The steps of the approach are outlined as follows:

a) Define the aims of the survey.
b) Identify the target population.
c) Choose the survey method.
d) Create the questionnaires.

e) Run a pilot survey.

f) Conduct the main survey.

g) Analyse the data.

a) To define the aims of the survey.

According to Burgess (2001), the initial task is to review the relevant literature of the target subjects. In this survey the target subject is to review the PDM systems as shown in Appendix C. This technology literature review will help the understanding of the PDM functionalities which will partially affect the design of the questionnaire. For example:

- In the *Utility Function*, there are two functions namely *engineer-to-order* and *design-to-order* which formed the bases of Question 1.
- In the *User Environment,* this will determine how PDM can be used along with other computerised applications as shown in Question 3.
- In the *User Function*, a PDM system can determine how product related data can be used within the product development processes, thus, the focused of Question 5 is being used to assess this situation.

The aim of the study was to investigate how companies dealt with the implementation of PDM. The number of companies surveyed was over seventy, although about twenty-six of them responded (Appendix D). The types of companies ranged from SMEs to large organisations, representing a cross-section of industries in aerospace, defence technology solutions, mining machinery manufacture, medical and pharmaceutical industries. The Department of Trade & Industry (DTI) classification of company size was used i.e. small firms (1 – 49 employees), medium firms (50 – 249 employees) to large firms (over 250).

b) Identify the target population.

The target population of this PDM survey was focused from the South and West of Yorkshire to the Northeast of England. The survey was carried out with the assistance of two members from the Agility Group (2004), at the University of Durham. The population of survey targets was selected mainly through the contacts of the Agility Group (Agility Group 2004) and the Applegate register (Applegate Directory 2002). The list of respondent companies is shown in Appendix D.

c) Choose the survey method.

There are different methods in surveying for a research. According to Oppenheim (1992) and Burgess (2001), the list of methods is face to face interviews, telephone survey, mail survey, email survey and web page survey. The email survey was chosen in the research. The reason behind this choice is based on two factors:

(i) The speed of response and,

(ii) Practically no cost involved.

d) Create the questionnaires

The questionnaires used for the surveys are shown in Table 3.1. The questionnaires were created based on brain storming sections with assistance from the Agility Group (Agility Group 2004). Apart from the considerations of the PDM commercial systems review. Questions 1 to 3 are relevant to the general applications of a company's computerised information systems in manufacturing, design, quality control and best practices. Several factors help to define the first three questions, these include the considerations of new product development time, production lead time, flexibility in responding to changes in demand volume and variety, quality and responsiveness to customer delivery requirements. Questions 4 to 9 are focused on surveying "why" and "what" are the benefits and disadvantages of implementing a PDM system to support their product development processes. The main factors which help to define the questions are the benefits, implementation and integration issues with their development processes in PDM technology.

	Questions	Select Relevant Application(s)
1	What is the company's Production System	a) Make to order (high variability, low volume and low inventory) b) Assemble to order (high variability, low volume and medium inventory) c) Make to stock (low variability, medium volume and high inventory) d) Others, please specify
2	What is the company's Manufacturing Control System	a) Manual b) Finite Scheduling Tools c) JIT (Just In Time) / Kanban d) MRP / MRPII e) ERP - Enterprise Resource Planning f) OPT - Optimised Production Technology g) Others, please specify
3	What are the computerised applications used in the company	a) CAD / CAE / CAM b) MRP / MRPII c) CNC / DNC Machines d) Quality Assurance System e) Product Data Management f) Finite Scheduler g) Electronic Document Management h) Electronic Data Interchange (EDI) i) Project Management j) Workflow Management k) Others, please specify
4	Who is your PDM provider?	Please Specify
5	In what circumstances did the company considered investing the PDM system	a) Improved Product Development b) Examples of successful companies from similar industry c) Increased sales e) Others, please specify
6	What is / are the company purposes of implementing a PDM system	a) To control part proliferation and establish standards b) To manage engineering drawing, search and retrieval c) To control engineering release and change management d) To manage product configurations e) Others, please specify
7	How long has the PDM system been in operation	a) Less than 6 months b) 6 - 12 months c) 13 - 24 months d) 2 - 3 years e) More than 3 years
8	What are the difficulties encounters of implementing PDM?	a) Lack of management support b) Lack of buy-in b) Due to working practice
9	What are the benefits gained from implementing PDM	a) Shorter product lead times b) Better quality and control of information c) Faster access and retrieval of correct information d) Better visibility of product development status e) Reduced unproductive engineering time f) Others, please specify
10	How much impact has the system contributed on the business in terms of competitive advantage	a) Considerably b) Moderately c) None d) Too soon to comment

Table 3.1: PDM Questionnaire Industrial Survey

e) Run a Pilot Survey

A pilot study has carried out among selected companies and within the School of Engineering at the University of Durham in order to refine the questionnaires.

f) Conduct the main survey

Prior to sending out the questionnaire, email and telephone invitations were made to prospective participants. The number of companies contacted was over seventy, of these twenty-six companies have completed and returned the questionnaires.

g) Analyze the data

The key results of the survey are shown in Tables 3.2, 3.3, 3.4 and 3.5. Table 3.2 shows that 20 companies out of 26 indicated that PDM had made considerable contributions, the other four said that PDM showed a reasonable return (moderately) and two of them were too soon to comment. Several organisations have defined PDM technology as a communication tool to its external vendors, as well as for document storage and retrieval. Some even made it their primary tool for communication within their product development teams.

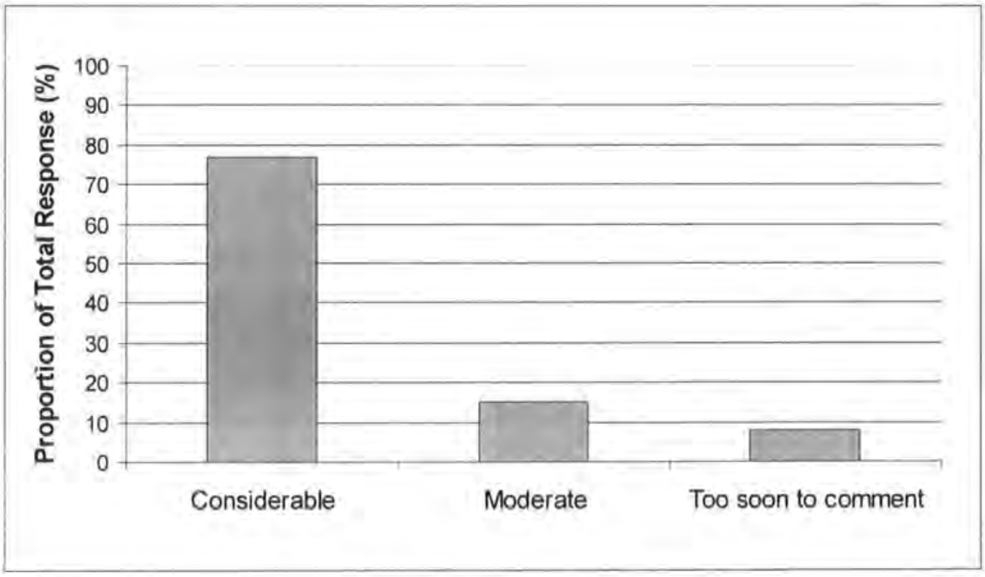


Table 3.2: Impact of PDM in Organisation

Table 3.3 indicates the key implementation difficulties captured by the survey. The three types of problems were the most common difficulties experienced by companies when implementing PDM. The result obtained from the survey has similar results as the one carried out by Whittaker (1999). In terms of unsuccessful PDM implementations, one of the most common reasons is "lack of top management involvement and support". Tachbrook (1999) identified the reason for the lack of management commitment to invest in PDM technology as being the fact that, from the top management's point of view, they only see the higher priorities i.e. invest in lower risk and higher return technologies. This is often due to the lack of understanding of the business activities which can benefit from deploying PDM technology. Another difficulty in implementing PDM was the lack of buy-in from users before they started using it. Finally, in terms of working practice, a global environment requires designing a system that will work in different cultures as well as different locations and supply chains, and this is far from easy.

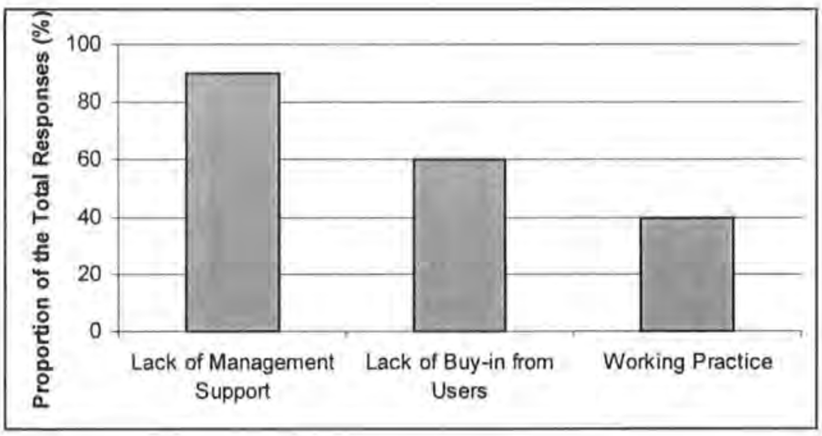


Table 3.3: Difficulties Encountered in the Implementation

Table 3.4 shows the main reasons why companies consider implementing PDM systems. Most of the companies in the survey used PDM to perform the following:

- managing engineering drawings,
- change management of engineering documents,
- managing product configurations, and
- design parts control.

There were also a few companies that use PDM technology for specific operations. One company commented: "The implementation was very fast, the system operated as required. Functionality and cost savings were being identified in the initial justification, taking longer than expected but was again successful."

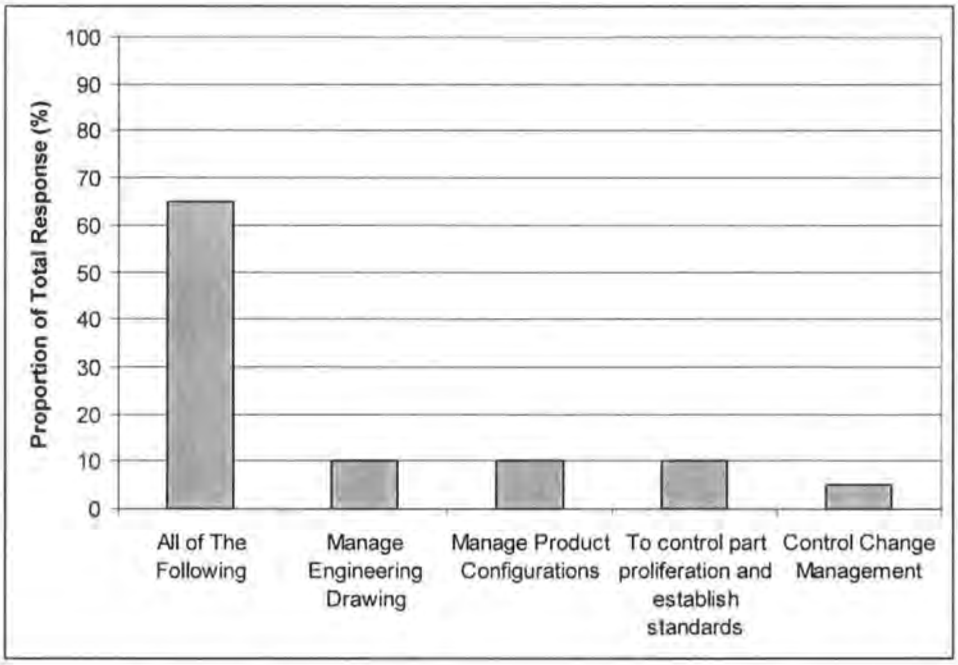


Table 3.4: Purpose of Implementing PDM

Table 3.5 shows the main reasons for investing in PDM systems. Improving the product development process is the key pre-requisite to invest in PDM, followed by successful implementation from other companies and, finally recommendation of top management. Improving the product development was seen to be vital for the investment of PDM systems. However, lack of top management involvement and support has resulted in many organisations failing to invest in such technology. Lack of management understanding may be caused by a lack of IT understanding in terms of its impact upon a modern engineering company. Senior managers that strategically organise their business should be aware that PDM is an expensive technology requiring significant capital and resources to gain benefits over time. It can only be considered as a priority over other projects when the extent of the strategic advantage that it can provide is clearly understood.

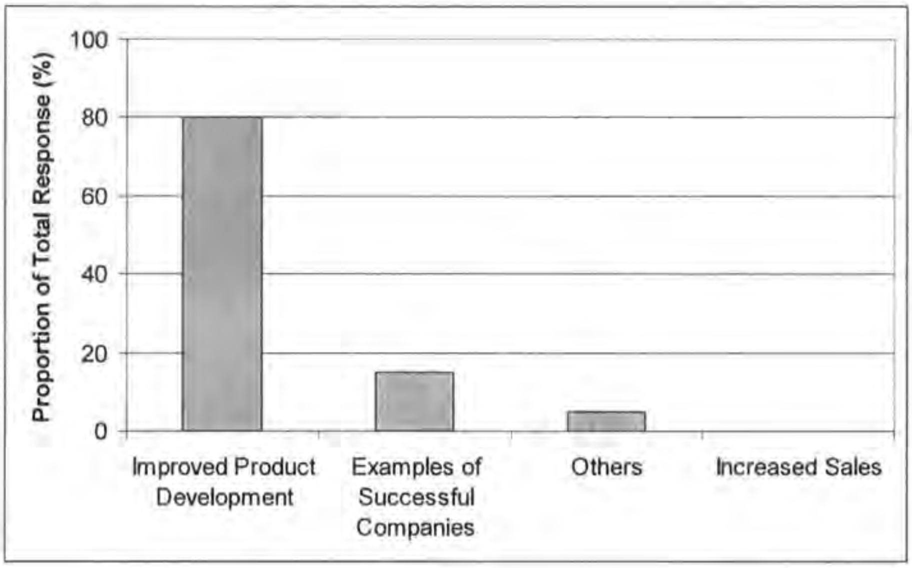


Table 3.5: Consideration for Investing PDM

3.4 SUMMARY

This chapter has highlighted the functionalities and the benefits associated with manufacturing enterprises. In addition, a brief review of an industrial survey based on the implementation of PDM system was also highlighted.

The PDM (or PLM) market is reaching the first stages of maturity. It is now expanding into the mid market having nearly saturated the largest spenders. Many small companies are beginning to realise the benefits of implementing PLM systems, but they are not willing to invest in the implementation due to their limited resources. Therefore a partial implementation of PLM systems is more likely, this would require less maintenance and customisation and thus resources. The largest companies can do extensive research before fully implementing a PLM system. The best solution for SME's would be a modular approach to PLM with a basic platform (e.g. basic PDM system) to which extra functionalities can be added at separate cost. This would give SME's exactly what they require in terms of specific functionality whilst keeping costs down.

The survey has shown considerable impacts for companies who have implementing PDM systems. In particular the survey has shown that the majority of the companies were using PDM systems to improve their product development processes. However, a

PDM system operating alone is not sufficient to achieve a truly collaborative and information distributive environment particularly at the early stages of product development activities as identified in the research hypothesis, and as such this survey has confirmed a new methodology is needed in order to bridge this discontinuity. The research will make use of other technologies and software systems to achieve this objective. This chapter provides the foundation of the PDM system which is the main element and backbone of the integration. The following chapter will describe the system overview on the development of the theory along with the implementation of various software components to form the product development integration architecture.

Chapter 4

SYSTEM OVERVIEW

4.1 INTRODUCTION

As stated previously in Chapter 1, the main objective of this research is to investigate methods for the effective management of the internet-based process of communicating new product requirements and manufacturing performance evaluations. The investigation is focused on the critical early stages of product development throughout the product life cycle using PDM and related Web-based technologies. A previous research project, CAPABLE Space (Bramall 2006) was mainly focused on developing a standalone experimental system. The approach of this research, however, is to explore different technologies and software systems to develop a collaborative and distributed manufacturing knowledge experimental system.

The previous chapters argued for a particular set of methods to be used in a collaborative product development environment. This chapter presents an overview of the system requirements and the implementation of the integration architecture and its associated software systems. The product development integration architecture consists of two domains namely design and manufacturing, inter-linked with two enterprise integration systems, ERP and PDM. In terms of the communication protocol, the integration environment is internet-based and uses the industrial standard data format XML and the data exchange mechanism XML Parser to support knowledge sharing and reuse. The approach of this methodology is not just unique, but also demonstrates flexibility in deploying current Web-based technologies, software tools and enterprise integration systems which can provide the foundation of an integration architecture to meet the demands of Web-based collaboration and distributed environment in a globally based product development scenario.

4.2 SYSTEM CRITERIA

In order to build a collaborative environment which enables the required product information to be distributed locally and externally throughout the enterprise to support product development, the system must fulfil a number of criteria (as illustrated in Figure 4.1):

1. Representation of product information from product design, processes and manufacturing capabilities (the CAPABLE system).
2. Knowledge-based system is needed to store design and manufacturing knowledge.
3. The terms used to represent knowledge must be defined to enable different personnel from different departments to understand them.
4. Information must be captured in an industrial standard format such as XML which can be shared and reused by different applications, for example, product information which includes design and manufacturing
5. The application of a Product Lifecycle/Data Management system to support information sharing.
6. The application of a process planning system to evaluate design alternatives which can provide automatic generation of process plans for alternative designs. Ideally, the system should support both aggregate and detail designs.
7. The application of an Enterprise Resource Planning system to analyse and evaluate capacity requirements planning to demonstrate and obtain resource scheduling
8. Finally, the development of a centralised client/server product development integration architecture and a hybrid P2P de-centralised network using open source technology.

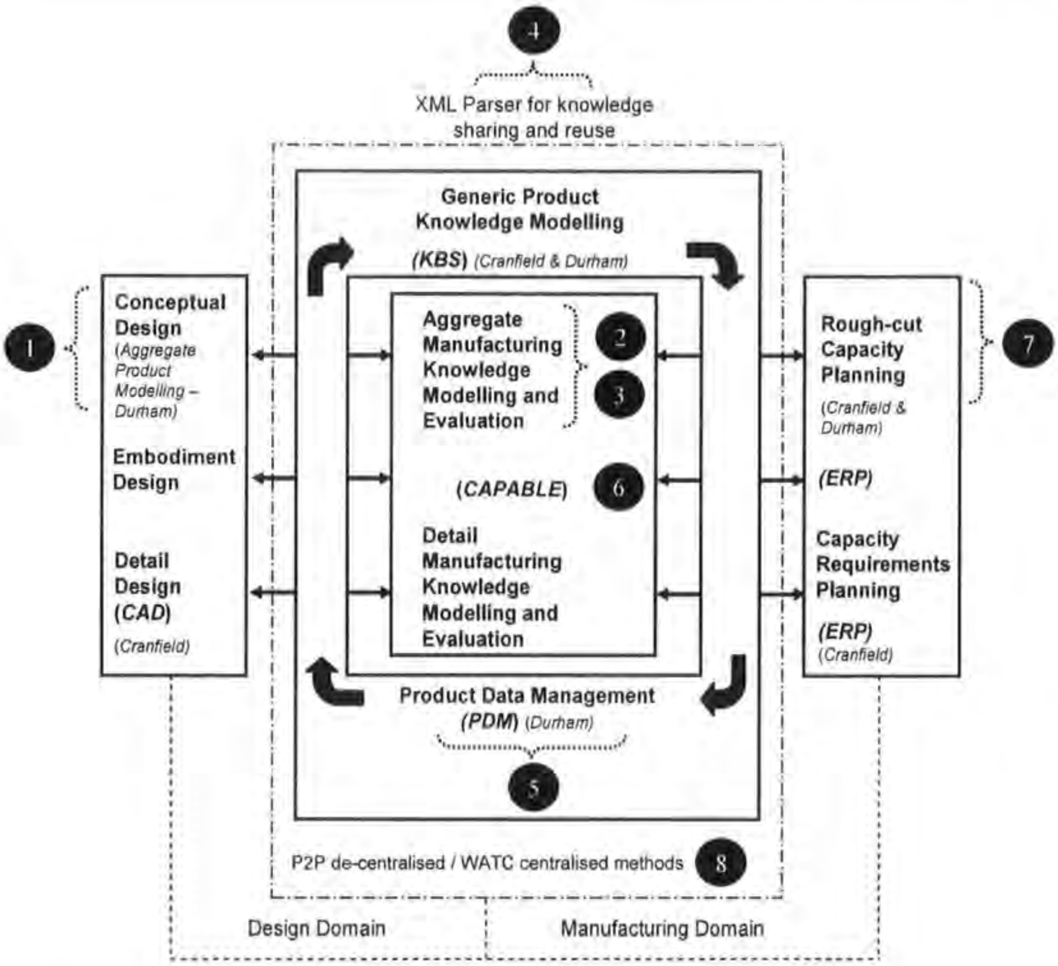


Figure 4.1: The Main Elements of the Proposed Distributed and Collaborative Product Development and Manufacturing Knowledge Management Environment (adapted from Maropoulos and Gao, 2000).

Overall, the novelties of the Durham contributions:

1. To allow the CAPABLE system to capture and store the industrial collaborators' production capabilities.
2. To enable the CAPABLE system to read and extract XML-based manufacturing knowledge.
3. The development of a generic manufacturing know-how data structure, which has been constructed as part of an organisational knowledge framework using an ontological approach and aiming toward capturing and reusing manufacturing knowledge. In particular, the effectiveness of decision making is increased.

4. Managing and co-ordinating the use of captured design and manufacturing knowledge within an enterprise. The captured knowledge can then be converted into an XML-formatted file and shared within a PDM system to support the product development process.
5. The introduction of an 'out-of-box' solution which provides an alternative framework and deploys this utilising a proprietary PLM system and open source P2P networking. The term 'out-of-box' means a solution is ready to be implemented without the needs of further customization of the software. The 'out-of-box' solution provides the benefits of rapidly reconfigurable network and minimal requirements for software customization to avoid system instability.

As the use of PDM systems for product design and development is becoming more widespread throughout the global manufacturing sector, the application of knowledge sharing, capturing and reuse using PDM systems is seen as being of primary importance for the competitiveness of the global enterprise of the future. The key benefit of adopting such an approach is the closer integration between the activities taking place during early design.

4.3 IMPLEMENTATION ISSUES

4.3.1 UML and Object-Oriented Techniques for Aggregate Manufacturing Models

The work of this part of the implementation is to enhance the capability of the aggregate manufacturing models (the CAPABLE System) in product features, manufacturing processes and production resources related to the industrial collaborators. The Unified Modelling Language (UML) (OMG 2003; Schmuller 1999) is used to model the structure of the information models together with the Java™ Object Oriented programming language for business logic implementation to create the prototype system. Details of the implementation are discussed in Chapter 5.

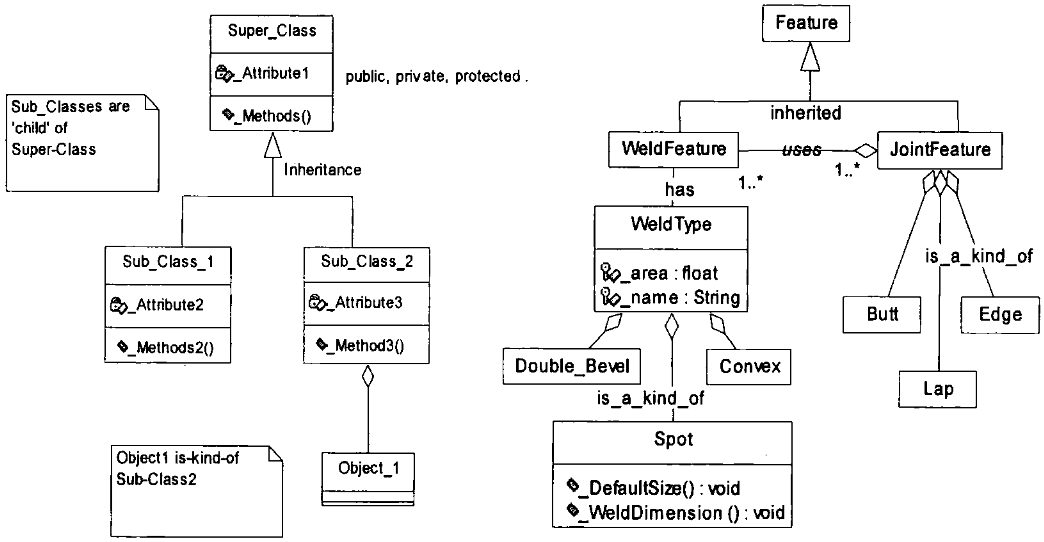
UML provides one consistent language for specifying, visualizing, constructing, and documenting object-oriented software systems, as well as for business modelling and has previously been used for enterprise modelling (Dorador and Young, 2000). The

UML is a holistic approach to systems modelling and includes additional expressiveness to handle modelling problems that earlier languages did not fully address.

Object-orientation is a well-established approach to managing complexity in computer programming. The use of classes in object-oriented programming offers a powerful way of representing the physical entities that are being reasoned about, their properties and the relationships between them. The overarching concept of object-orientation is that of abstraction, which concerns the level of complexity modelled by the system and allows the programmer to ignore those aspects of the system which are irrelevant and concentrate on the important factors. A powerful way of managing abstractions is through the use of hierarchical classifications. There are three attributes of object-oriented programming languages and these are:

- Encapsulation - is the mechanism which implements information hiding and modularity (abstraction).
- Inheritance - is the process by which one object acquires the properties of another object further up the hierarchical classification. New classes and behaviour based on existing classes to obtain code re-use and code organisation.
- Polymorphism - is a feature that allows a single interface to be used for a general class of actions.

Object-oriented programming enables manufacturing models to be constructed in modular fashion as shown in Figure 4.2(a). Figure 4.2(b) depicts an example of a superclass 'Feature' which consists (*inheritance relationship*) of two subclasses namely 'WeldFeature' and 'JointFeature'. Further down the class taxonomy, 'WeldFeature' *has* (*association relationship*) 'WeldType', whereby, 'Double_Bevel' and 'Convex' *is_a_kind_of* (*aggregation relationship*) 'WeldType'. Similarly, 'Butt', 'Lap' and 'Edge' *is_a_kind_of* 'JointFeature'. Each of these may then 'inherit' properties from their class, such as the attribute which define the *_area* in the 'WeldType' *(concept of encapsulation)* class and, hence, 'WeldType' can use a specific type of welding to calculate the welding dimension such as the method provides by the subclass 'Spot', *_WeldDimension ():void. (concept of Polymorphism)*.



(a) Object Oriented Programming Theory (b) Implementation

Figure 4.2 Example of Object-Orientated Method in UML Class Modelling

4.3.2 The Organisational Knowledge-Based System

An organisational knowledge-based system for capturing and representing manufacturing know-how has been developed using an ontological approach. The important aspects of the implementation are the adoption of an axiom-controlled ontology and, the applications of Web-based technology including RDF and agreeable vocabularies to define the semantics and the meaning of terms. Further particulars of the implementation to create this ontological's knowledge system are discussed in Chapter 6. The captured knowledge is converted into the industry-standard eXtensible Mark-up Language (XML) and then shared within a web-centric Product Data Management (PDM) system to support a collaborative and distributed product development environment which provides the benefit of closer integration between the activities taking place during early design.

4.3.3 Product Development Integration Architecture

In general, the integration environment can be distinguished into *three layers*. The first layer is the enterprise system which incorporates PDM and ERP technologies. The second layer is the communication and data exchange mechanisms, i.e. other peers, JuxtaPose (JXTA), client/server and XML Parser. The third layer consists of the Manufacturing and Design Domains to support product development processes.

A Peer-to-Peer (P2P), decentralised communication network is used to evaluate how such a hybrid integration can enhance the integrity of knowledge and data sharing and the efficiency of network communication for collaboration of smaller companies within the supply network. JXTA is a proposed method for open source network connectivity (Cheung *et al.* 2004; Aziz *et al.* 2005). The researchers have extended and further developed the product development integration architecture with a new framework to explore alternative computing power and network bandwidth in order to reduce cost as well as providing capabilities to smaller companies for the autonomous of sharing information with larger co-operations. This new proposal will be discussed in Chapter 7.

The implementation issues of the various components to form a collaborative and distributed network are highlighted in the follow section. However, the discussion of implementing STEP Modeller and the Design Knowledge-Based System are based on the research work at Cranfield University (Aziz *et al.* 2004).

4.3.3.1 XML Parser

A XML Parser is the piece of software that reads XML files and makes the information from those files available to applications and programming languages, usually through a known interface like the DOM (Document Object Model) or SAX (Simple API for XML parsing) (Holzner, 2001). The XML Parser is responsible for testing whether a document is well-formed and, if given a DTD (Document Type Definitions) or XML schema, it will also check for validity (McLaughlin and Loukids, 2001).

XML Parsers can be used to reconstruct instances of Java classes from the XML data (see Chapter 7). There are two different methods of implementing XML Parsers i.e. DOM or SAX. DOM views an XML document as a tree structure and loads the entire document into memory. It builds parent-child relationships between nested elements. The DOM API provides standard methods for querying the XML document and reconstructing Java objects. SAX is an event-based parser that "fires" different events based on the element parsed. A SAX parser, unlike the DOM parser, does not maintain a default model for the parsed data. If a listener can be set on the SAX parser to listen to parsing of specific tags, then whenever the corresponding element is parsed, the listener can construct a Java instance of the class corresponding to the element.

For implementation efficiency, SAX is the equivalent of a depth-first, LR (linear recursive) traversal of a tree; a tree can be generated from the events. SAX makes it possible to process a very large document using relatively constant resources, but makes arbitrary queries and transformations difficult; DOM makes queries and transformations easy, but resource usage increases with the size of the document. For system resource usage, SAX allows the user to parse large documents without having to store all of the data in memory, while DOM generally must build the entire tree for the document in memory.

In terms of validation concerns, DOM permits users to make changes to the data, while SAX is a read-only system. Validation includes analyzing, validating and processing XML schemas and resource descriptions. The parser analyses syntactically the statements of a given Schema / XML file according to the specification. The parser checks whether the statements are contained in both XML schemas. The implementation in this work uses the method of DOM to implement the XML Parser due to the fact that all the information in the organisational knowledge-based system is being transferred into a document based XML file.

4.3.3.2 PDM /WATC

A PDM system was one of the main technologies this research deployed before it evolved to the current PLM solution. In general, a PDM system provides different functionalities as shown in Appendix B. However, the research work described here makes use of 'Change Management', 'Workflows', 'Lifecycle' and 'Document Manager' functionalities. Part of the implementation is to make use of the PDM system as an 'integration wrapper' which provides the ability to "wrap data and knowledge" from different domains into a common format, such as XML, so that a XML file can be shared within a distributed PDM environment.

A Workflow Activity Task Controller (WATC) methodology built upon PDM technology has been defined and developed to distribute product information and knowledge in a collaborative product environment (see Chapter 7). The implementation of WATC is centred on the "lifecycle" and "workflow" functionalities of the PDM system and the application of a Java-based XML Parser. "Lifecycle" function defines the timing of the development stages and "Workflow" function determines what processes and the interactions take place at each stage.

4.3.3.3 ERP

The Compiere ERP (Compiere CMPCS for manufacturing) is an open source software and by definition it is free. The application of this freeware in capacity requirements planning for the work is described in Chapter 8. Compiere CMPCS covers all manufacturing activity within various types of production environments. The main modules are Resource Planning, Production Plan & Demand Management, MPS, MRP, CRP and Manufacturing Orders & Repetitive. In order to use the software more effectively the user must customize the system to suit the needs of the work being undertaken.

One further aspect of the software is that apart from the *import* function, which can be used to download the process plan, the majority of the information was manually inputted. The system is not completely developed yet, and its performance over low speed dial-up connections is very poor (however the web interface is usable). Also, it cannot be deployed with another open source database as yet.

4.4 SCOPE OF INDUSTRIAL APPLICATIONS

4.4.1 Background of Industrial Applications

The main focus of this work is to capture two different kinds of products to prove the flexibility of the demonstration system and prove the hypothesis of the project objectives. This novel method has been tested with real products and processes provided by the collaborating companies. The research addresses the needs of a very large section of industry. The research is particularly relevant to manufacturing environments with small batch sizes, medium to high product complexity, increasing product variety and decreasing product life-cycles. The inherent ability of the proposed methods allows early design/concept evaluation within a distributed environment, and could improve the responsiveness and quality of quotations, which is vitally important if supply chain companies are to take full advantage of e-commerce opportunities.

ArvinMeritor and Mabey & Johnson Ltds were selected from the manufacturing industry to participate in this research. This selection was based on companies operating internationally with globally distributed supply networks. Both companies appreciated a need to strengthen their responsiveness to the customers' requirements, in particular at the early product development stages. This will be achieved by bridging the communication of design and manufacturing through PLM/PDM. The new method will also provide the companies the opportunities to enhance their business through distributed sourcing options. They have foreseen that this research will support the infrastructure of how PLM/PDM technology can be used to deliver their key business objectives. Furthermore, the design and manufacturing software vendors are also interested in this research, in particular PTC who is enthusiastic of the new method developed in this research to enhance the effectiveness of applying PLM system.

4.4.2 Outline of the Application Scenario

A UML Sequence Diagram (Schmuller 1999) as illustrated in Figure 4.3 outlines the application of the product development integration architecture in a Web-based data interoperability scenario between the design and manufacturing domains and the enterprise systems. The overall system functions as follows:

1) Create XML-based Knowledge file to be re-used
   a) Create instances within the Knowledge based-system
   b) Save instances in XML format from Protégé2000
2) Load the XML document
   a) Invoke the PDM system and use the 'Create Document' and 'Check-In' functions to load the XML file into a PDM cabinet
3) Use the XML-based Knowledge file
   a) Invoke the PDM system and use the 'check-out' function to load the XML file into the user's local web-server.
4) Invoke the third party software system
   a) Run the CAPABLE Aggregate Process Planning System.
   b) Invoke the XML Parser from the CAPABLE to associate with either the Product Features or Resource (Factory Modelling) for extracting the XML file.

5) Save the process plan
   a) Use the native functions provided in CAPABLE to generate process plan output as a text file.
   b) Use the 'check-in' function to load the text file into a PDM cabinet
   c) 'Check-out' the plan for Capacity Planning using an ERP system

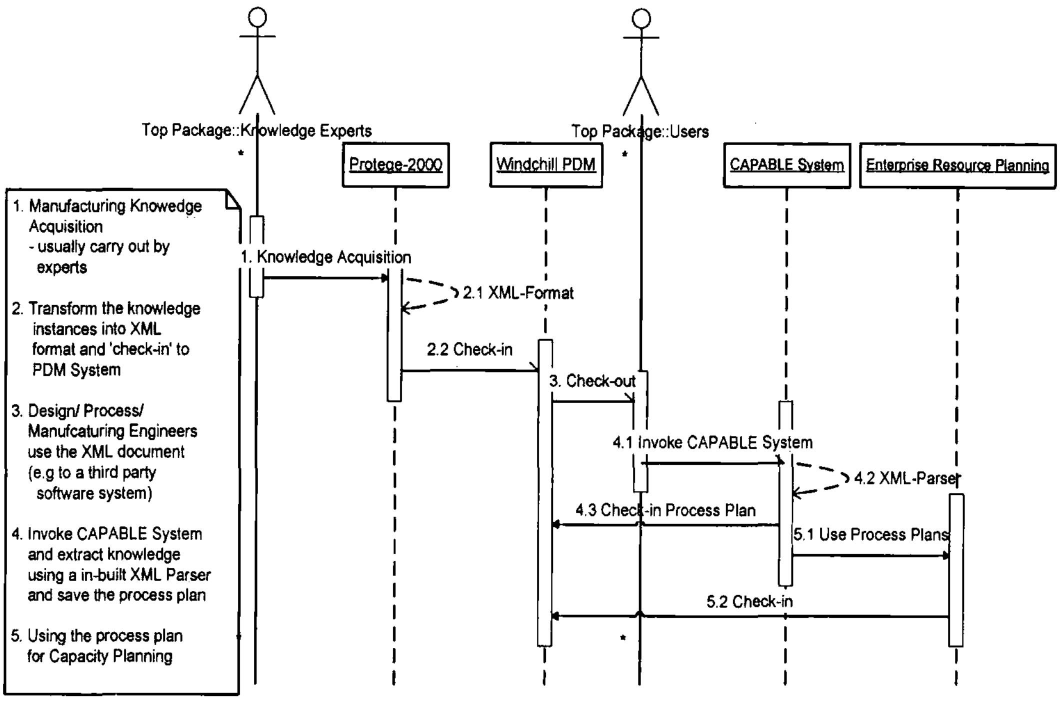


Figure 4.3: Systems Applications in UML Sequence Diagrammatic Representation

4.4.3 The Bailey's Bridge Panel

Figure 4.4 illustrates a complete Bailey bridge. The concept of Bailey Bridging was developed during the 2nd World-War by Sir Donald Bailey to enable bridges of varying spans and carrying capacities to be speedily erected manually, by unskilled labour. Mabey & Johnson Ltd recognised that the original requirement of an assault bridge limited the wider application of Bailey bridges and in 1974 introduced a panel bridge designed for long term permanent applications and was designed to carry greater loads and traverse spans of up to 80m and carry up to 3 lanes of traffic.

Figure 4.5 illustrates a single Bailey bridge panel. The bridge panel was constructed from 14 steel beams (4 horizontal, 8 diagonal and 2 vertical), 6 different steel plates and 16 square slots. The overall dimension of the bridge panel is approximately 3 m by 2 m with a weight of 130 kg. Assembling the bridge panel is a complex task. The number of welding processes required to assemble the panel takes more than 240 procedural steps. The process model covering alternative manufacturing and assembly processes is created from real process data provided by the collaborating company. The type of processes used by the company include 'Robotic Welding', 'Manual Welding', 'Drilling' and 'Galvanising' and so on. For alternatives, several joining processes are available to the optimising routines. Factory Models consists of different machines such as 'Galvanizing Trailers', 'Surface Coating', 'Robot and Manual Cells' and 'Drilling'.

The industrial collaborator developed new products, one of them is a 'floating platform' that uses the embankment for transportation was designed using the concept of Bailey bridging. As the new Bailey's bridge design is becoming more complex, so are the requirements of the manufacturing capabilities. Thus, this is one of the main reasons that the industrial collaborator is interested in adopting new technologies, particularly the use of a centralised network in association with PLM and ERP systems, as well as the applications of design and manufacturing knowledge management techniques in a Web-based environment.

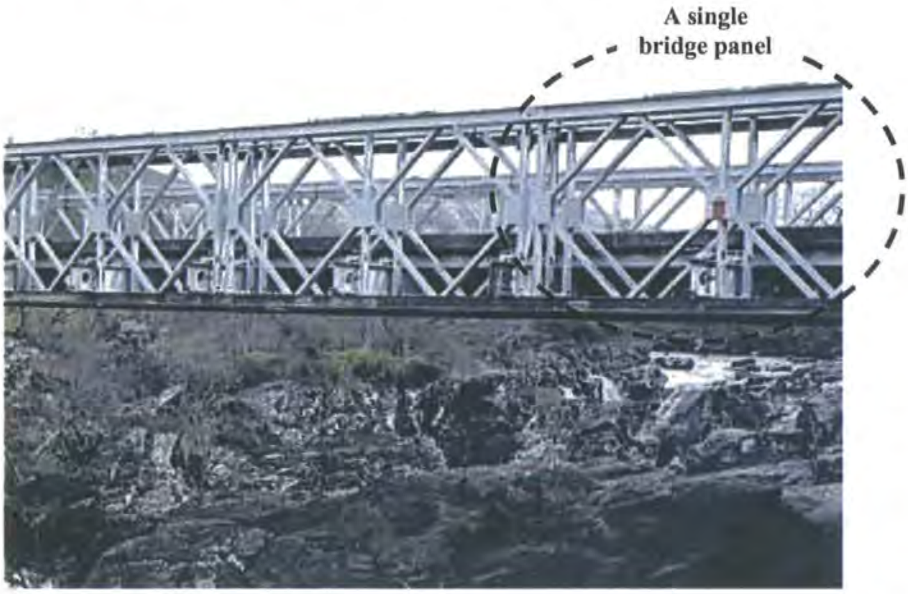


Figure 4.4: Bailey's Bridge – *(Picture courtesy of Mabey & Johnson, UK)*

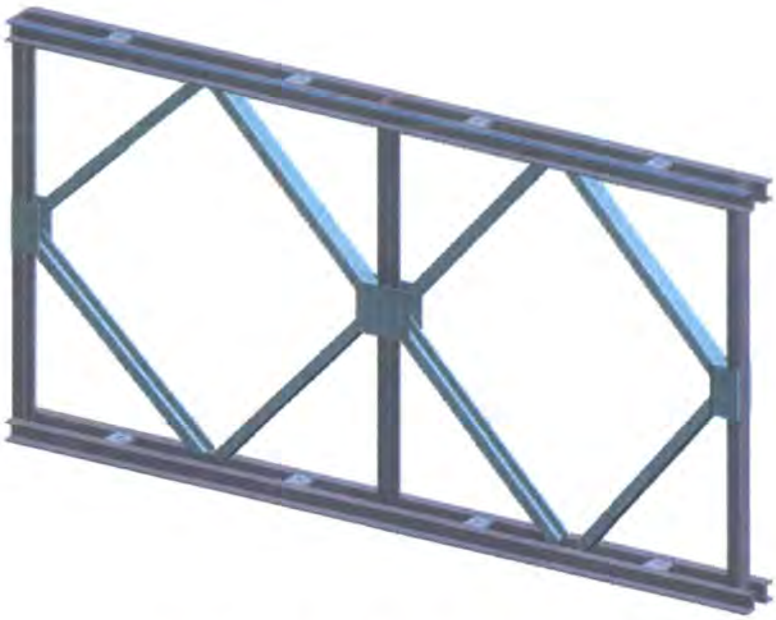


Figure 4.5: A Single Steel Bridge Panel

4.4.4 The Vehicle-Door-Latch Assembly

The second case study product is a vehicle-door-latch construction design. A completed vehicle-door-latch contains more than eighty individual parts. In order to simplify the test, it was assumed that the general assembly of a single vehicle-door-latch was made up of five individual components: the retention plate, pawl, pawl rivet, claw and claw rivet as shown in Figure 4.6. The way to assemble the latch was by using the *orientation* assembly process supported by *feeding equipment* as modelled in the CAPABLE System, which is discussed in Chapter 5. Figure 4.7 shows a light vehicle door module example, which contains a vehicle-door-latch. The overall objective of the case study using the vehicle-door-latch is to demonstrate the advantages of a de-centralised product development network scenario. Furthermore, the root of the design is initiated by a customer requirement specification. In this example, a complex requirement from an automotive OEM is used. One of the main reasons to use this demonstration scenario is that the majority of the industrial collaborator's design is outsourced to local suppliers. Apart from the interest of utilising PLM technology to enhance design knowledge, the industrial collaborator is also interested in adopting a decentralised network to do business with smaller companies. The aim is to prove that the use of P2P technology is more economically feasible and be able to control their own knowledge for smaller enterprises to form a collaborative network with larger companies.

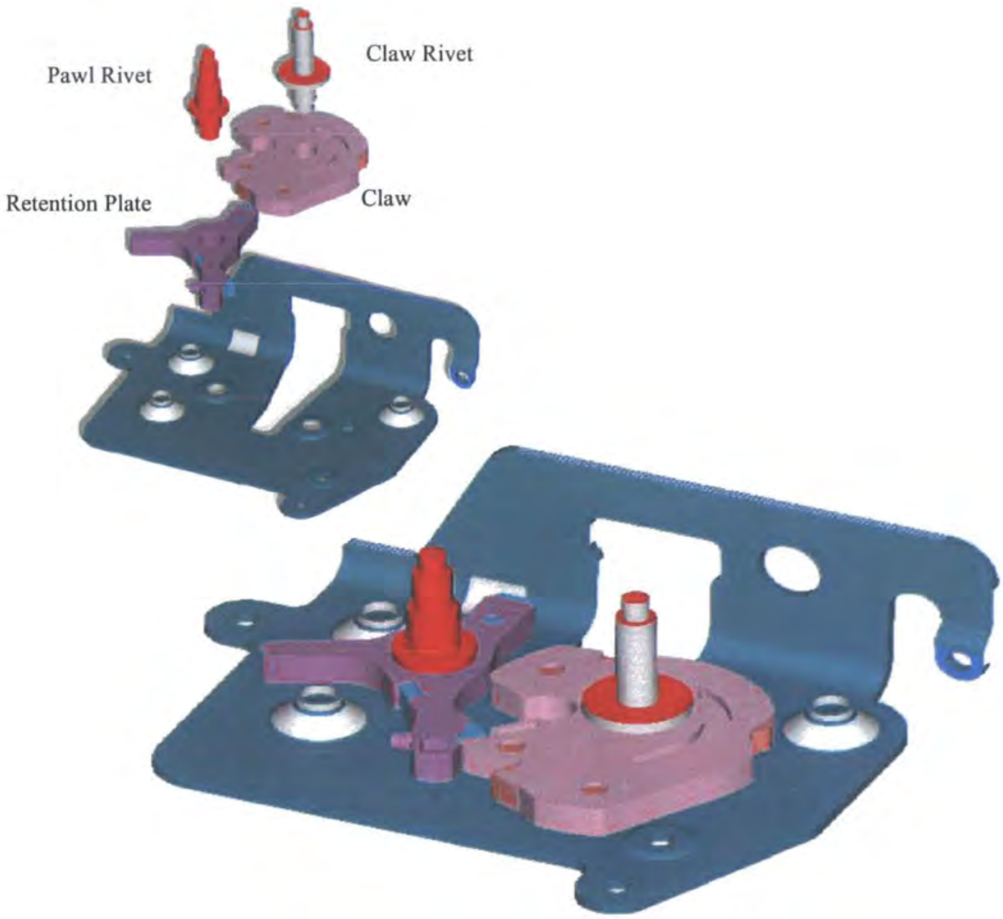


Figure 4.6: A Simple Vehicle-Door-Latch Assembly

Figure 4.7: Light-Vehicle Door Module – (*Picture courtesy of ArvinMeritor UK*)

4.5 SUMMARY

This chapter has outlined the scope, the functionalities and the application of the proposed "distributed and collaborative product development and manufacturing knowledge management" environment within the context of the research hypothesis. The system criteria and novelties of the Durham contributions have been defined. This chapter has also presented the main theme of this research which provides a preview of the material covered in Chapters 5, 6, 7 and 8. In general, the integration architecture comprises:

- The Design Domain for capturing design based data and rules, the use of STEP modeller for design features.
- The Manufacturing Domain consists of the Manufacturing Know-how KBS for qualitative and quantitative manufacturing based knowledge, the CAPABLE / Space System for aggregate process planning.
- The deployment of Web-based systems and technologies such the PDM and ERP as well as the standard data exchange format.
- A hybrid integration methodology as a subset of the overall integration architecture.

The detailed industrial testing of the proposed methods are documented in Chapter 8. In the next chapter, the focus of the discussion will shift to the development of aggregate manufacturing models (the CAPABLE System). The CAPABLE System is an early process planning system, used to demonstrate the reuse of manufacturing knowledge at the early stages of communicating design concepts and potential manufacturing scenarios.

Chapter 5

THE AGGREGATE MANUFACTURING MODELLING

5.1 INTRODUCTION

The CAPABLE System, an aggregate manufacturing model, is one of the key components in the product development integration architecture as highlighted in Chapter 4. As previously described in the literature reviews Yao *et al.* (1998) developed a CAPABLE Welding System to evaluate the design of sheet metal fabrication at the early stages of product development. Bramall and McKay (2001a and b) developed a CAPABLE Space system for the aerospace applications. This work, the distributed product development project, however, is primarily to further develop and make use of the CAPABLE Space system and adapted the capabilities of the CAPABLE Welding System in welding features and welding process modelling.

The following sections will focus on describing the additional development of the CAPABLE System and the aims are:

- To allow the system to capture and store the industrial collaborators' production capabilities.
- To enable the system to reuse XML formatted knowledge statements.

The reason for this development is the need to have a modelling and planning ability to deal with case studies of the industrial collaborators.

5.2 APPROACH TO THE AGGREGATE MANUFACTURING MODELLING

The research has taken the approach of developing the theoretical aspects and development of aggregate product, process and factory resource models that facilitate capturing key early design data needed for planning and evaluating production using quality, cost and delivery criteria (Bramall *et al.* 2003; McKay *et al.* 2003). This has led to the theoretical definition of weld features in the aggregate product model and of welding and non-shape-changing process classes for the aggregate process model.

An "aggregate weld feature" is made up of a permissible combination of a "weld type" and a "joint feature" as illustrated in Figure 5.1. The weld type class contains 14 types of welds and the joint feature class contains 6 types of joint features. The combination of these weld types and joint features can satisfy the definition and representation of the vast majority of the weld features found in industrial products. The aggregate welding process class includes four types of welding processes namely, gas welding, arc welding, electron beam welding and laser beam welding. The non-shape-changing aggregate processes include surface coating, galvanising and surface preparation which are frequently deployed in welded fabrications.

The research work has also extended the assembly process models with a special class of "orientation operations", frequently deployed for manual or automatic assembly. The new aggregate models have been formalised using the UML and have been integrated with the aggregate models.

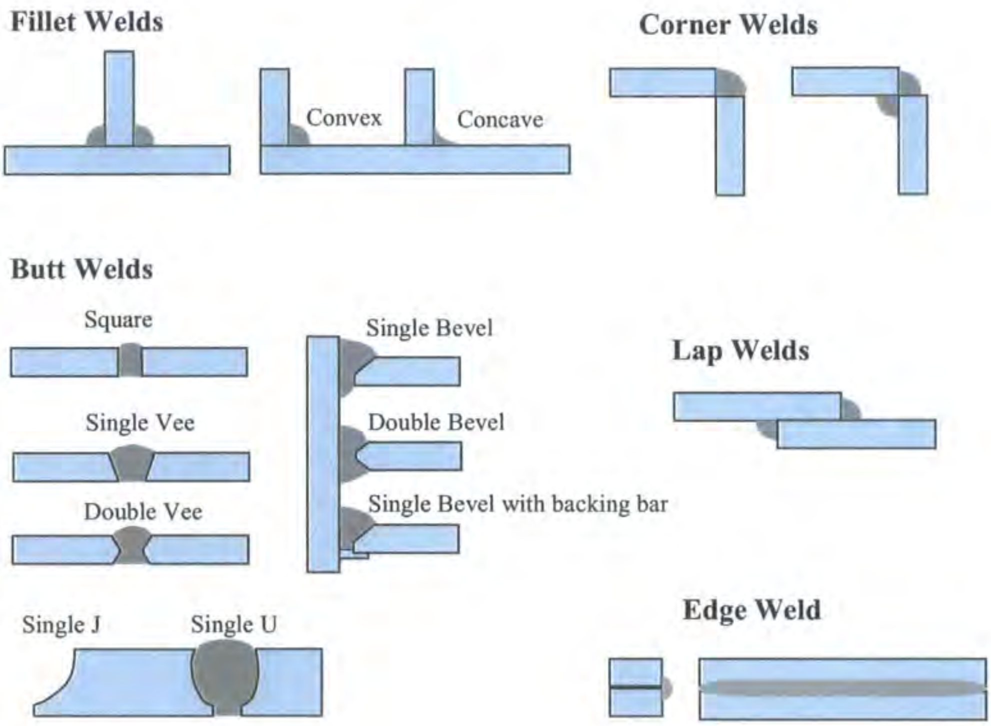


Figure 5.1: Example of Weld Types and Joint Features (BS EN 1708-1:1999 / BS EN 1708-2:2000 / BS EN ISO 17659:2004)

5.3 DEVELOPMENT OF THE AGGREGATE MODELS

The initial task of the implementation was to translate and model the CAPABLE system into the UML structure as shown in Figure 5.2. The UML representation consists of different *packages*. Within each of the *packages*, it consists of a string of different *class* representations to determine the way to encapsulate relevant data of a specific domain, for instance, a process model represents the data encapsulation of process capabilities. The Resource Model captures the data of factory resources and the Product Model captures product features. *CAPABLEObject* is the highest level of the system, which supports all major data models such as resource, process, process planning, product models and knowledge model. In addition, *CAPABLEObject* uses Java object serialization (Java Software 2006) to identify the type of object that is being dealt with and holds the path of the directory where objects of this type should be stored in the databases. This research was to modify and further develop the CAPABLE System with additional process capabilities, resource requirements and product features as shown in Appendix E. The additional work is discussed in the following sections.

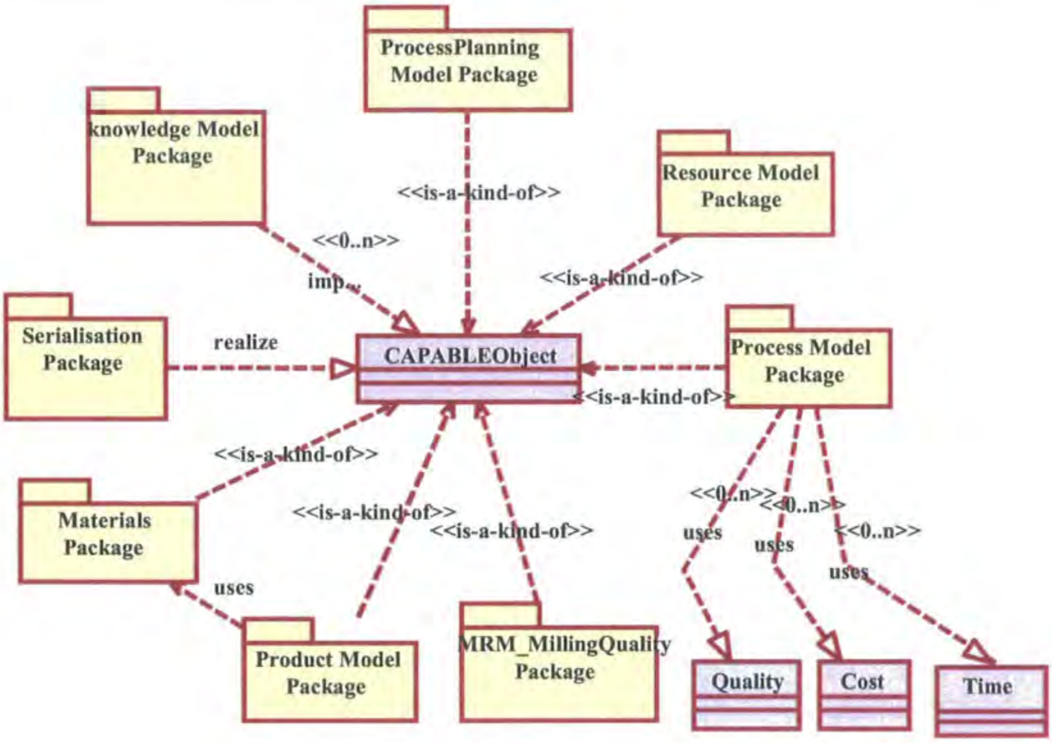


Figure 5.2: Overall UML Class Representation of the CAPABLE System

5.3.1 Development of the Product Model

The additional development of the product model was focused on the introduction of welding types and joint features for the top-level Product Model of the CAPABLE System as illustrated in Figure 5.3. The product model uses object-oriented methods (Zhao *et al.* 1999; Bramall 2006) for the representation and management of a multi-level product structure, comprising products, components and features as objects. The additional classes allow the CAPABLE system to support the conceptual design of the Bailey's steel bridge panel in a form that is suitable for integration with the process planning system. According to Bramall *et.al.* (2003) and McKay *et.al.* (2003), the early stages of design are concerned with defining product function and overall structure based on positive features and structural joint features with additional tolerances. Feature-based design techniques can support these active product representations and are an effective medium for transferring information between design and process planning, forming the basis of the product model (Shah and Mantyla 1995).

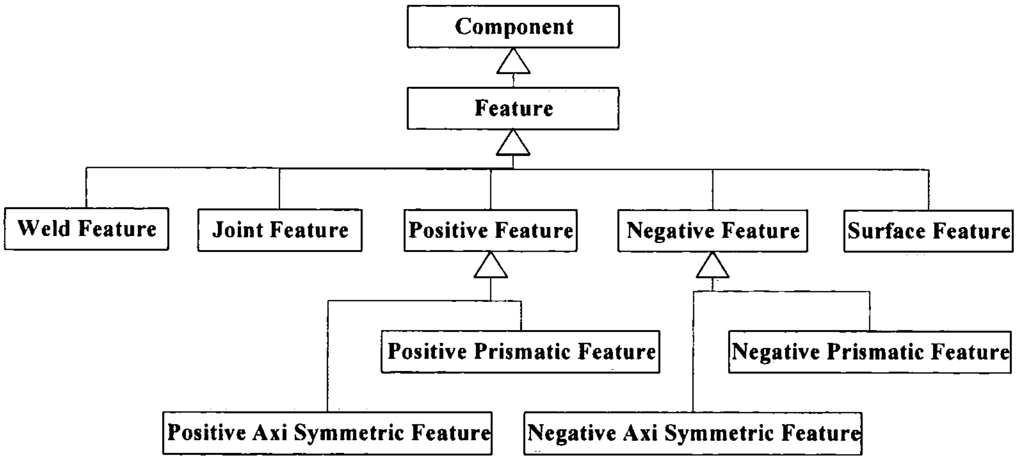


Figure 5.3: Top-Level Product Model Class Diagram

Figure 5.4 depicts the UML class diagram of the product model representation in welding. *Feature* is the super class of *weldfeature* and *JoinFeature*, where *weldfeature* consists of sub-classes to represent different weld types. For example *DoubleU, Double_Vee, Square, Convex and, Concave* and so on, the terms used in the class diagram are consistent and corresponded with industrial standard as shown in Figure 5.1. In terms of *JointFeature* there are six standard types which are widely used in the welding industry namely *Tee, Corner, Plug, Butt, Lap* and *Edge*. In the application of welding assembly of a particular member of the *StandardBridgeM_J* panel. *JointFeature*(s) is normally associated with a set of *WeldFeature(s)*. For example, *JointFeature* of *Butt uses WeldType* of either *Double_Vee* or *Single_Vee* or *Square* or combination of all which is denoted by (1..*). Similarly, *JointFeature* of *Corner uses Convex* or *Concave* or combination of the two.

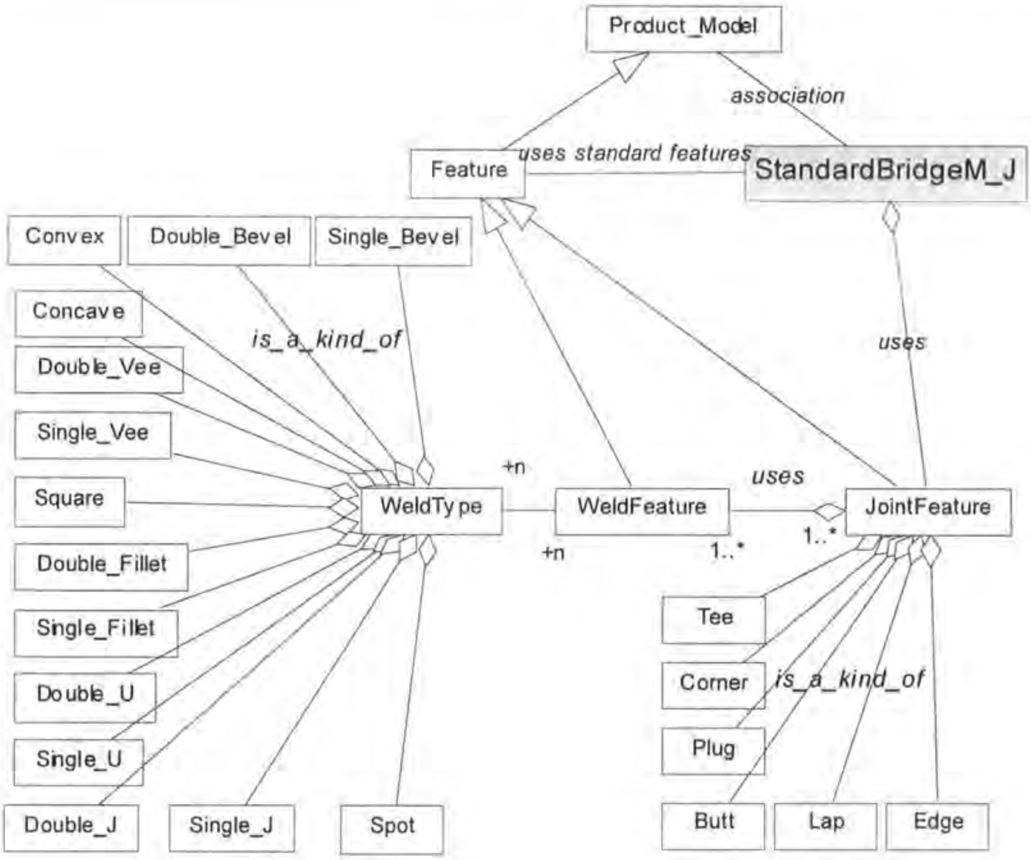


Figure 5.4: Additional Feature Classes for the Bailey's Bridge Product Model

5.3.2 The Process Model

Bramall *et al.* (2003), states that "Manufacturing is characterised by two kinds of activities; discrete parts production and the subsequent assembly of these parts to generate the finished product". The hierarchy of process models for discrete parts manufacture broadly follow the top-level classification presented by Allen and Alting (1986), which groups processes according to their morphological characteristics. Two broad categories of processes exist within this classification, shape-changing (or mass reducing) and non-shape changing (mass conserving). An object-oriented approach has been adopted to implement the process models, creating a hierarchy of classes as illustrated in Figure 5.5, which are used to model the different types of process at various levels of abstraction. For each process class which can be instantiated, the aggregate planning system has a set of functions which calculate process times based on the attributes of the selected feature and resource.

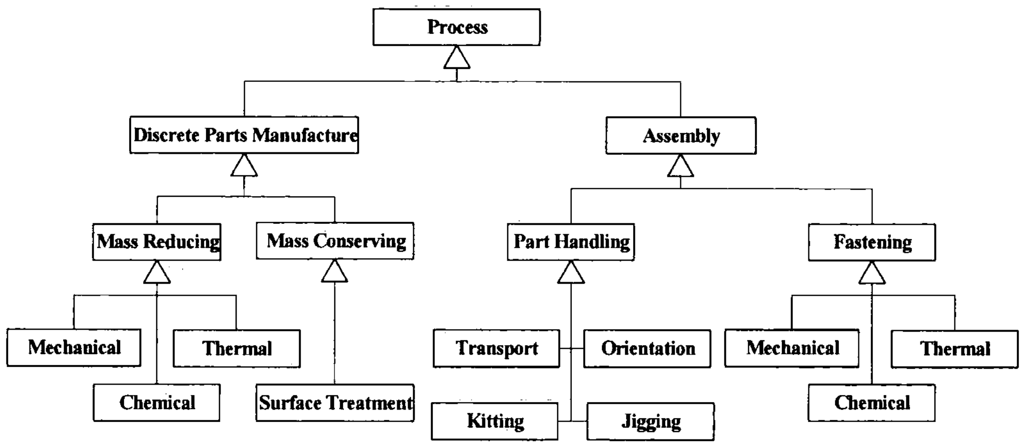


Figure 5.5: The Top Level Process Model Class Hierarchy

5.3.2.1 Development of the Process Model

The illustrations of Figure 5.6, 5.7 and 5.9 depict further additional processes which were not available in the CAPABLE System. The diagram represents the different type of processes which are specifically used to model the manufacturing capabilities of the industrial collaborators. Figure 5.6 illustrates the additional classes for *MassReducing* and *MassConserving.* Within the *Mechanical* hierarchy, processes such as *Shearing, Nibbling* and *Sawing* were introduced. These classes are used to model the capability of steel-beam preparation such as reduction of beam length and removal of a small part of the material on a flat surface, for example a small hole. Within the *SurfaceCoating* and *SurfacePreparation* of the *MassConserving* class, several sub-classes were implemented to model non-shape changing processes such as *SC_Galvanising, SP_Greasing, SP_Degrease* and *SP_ChemicalCleaning.* In particular, *SC_Galvanising* is a hot dip surface coating process which is widely used in zinc coated structural steels to prevent corrosion and expand its lifecycle, an application is the Bailey's steel bridge panel. *SP_ChemicalCleaning* is the process to apply chemical cleaning prior to the application of galvanising process. *SP_Greasing,* and *SP_Degrease* processes in this work are mainly used for the applications of preparing surface coating of discrete small parts for the assembly of the vehicle-door-latch.

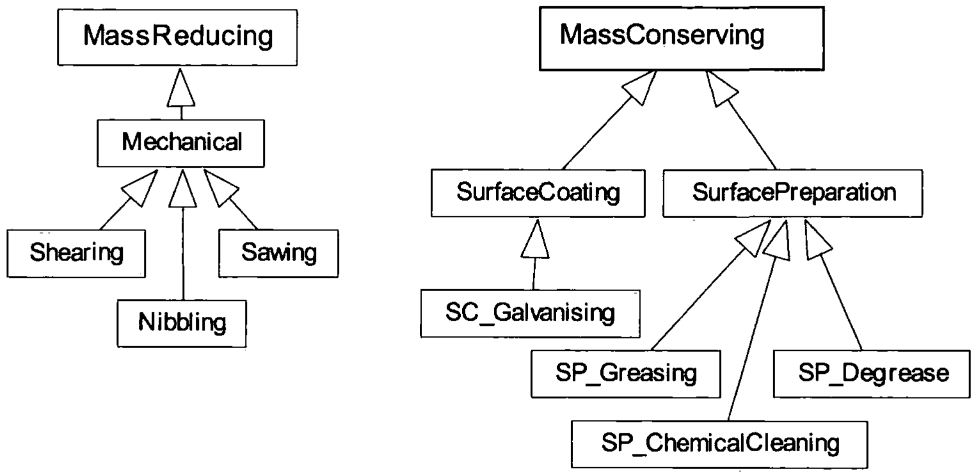


Figure 5.6: Additional Classes for *MassReducing* and *MassConserving*

Figure 5.7 illustrates the additional classes for *Discrete_Parts_Manufacture* and *Part_Handling.* This work has introduced two additional classes namely *MaterialForming* and *Orientation. Bending* is a subclass of *MaterialForming* which is used to form metal shape, for example, U-shaped steel beam. Orientation has a group of subclasses to capture data of handling the sub-components of the vehicle-door-latch assembly. Other processes which are used to support the assembly sequence of the vehicle-door-latch are *OR_Feeding and OR_Transferring* to ensure the units are securely fed. Figure 5.8 illustrates example attributes and operations of the *Orientation* class which has sub-classes of *OR_Feeding* and *OR_Placement.* Generally, *Orientation* is mainly used to calculate the time required for assembling certain product features in the modelling environment.

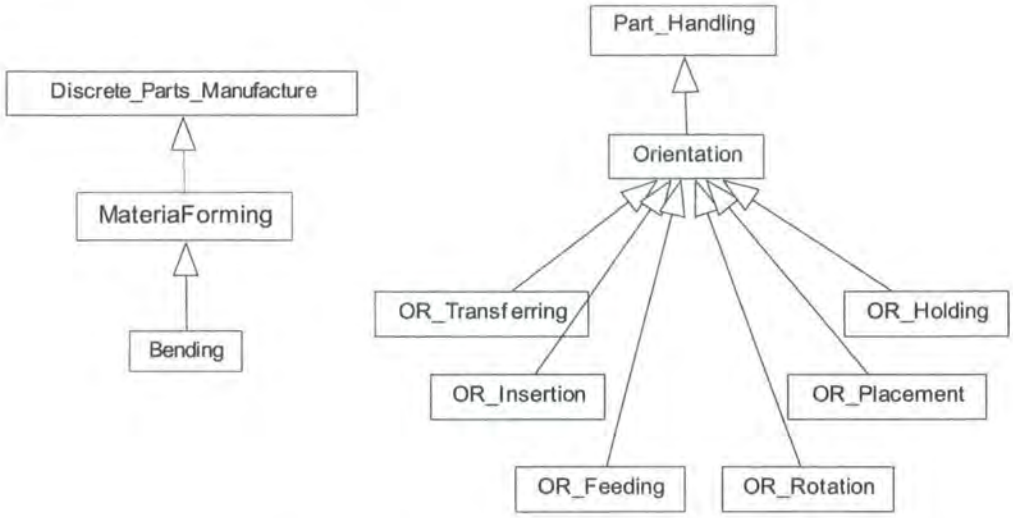


Figure 5.7: Additional Classes for *Discrete_Parts_Manufcature* and *Part_Handling*

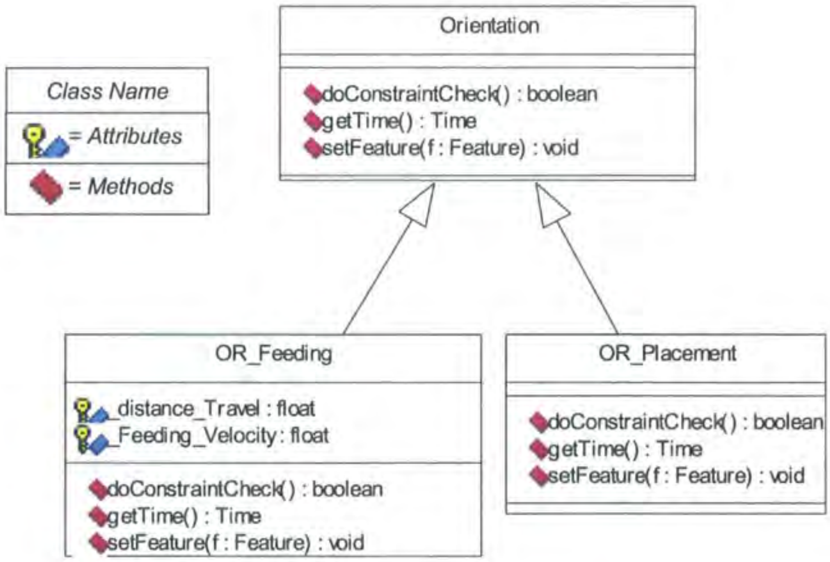


Figure 5.8: Selection of Attributes and Methods in *Orientation* Classes

Figure 5.9 illustrates the additional *welding* class in the *Thermal* joining process. Within the *Welding* class there is a set of dedicated welding processes such as *GasWelding, ArcWelding, ElectronBeamWelding and LaserBeamWelding.* This part of the research work is particularly concerned with capturing the capabilities of *ArcWelding* which will be used to model the making of a Bailey's single steel bridge panel. Figure 5.10 shows example attributes and operations of the *Welding* class which has a subclass of *ArcWelding*. *GasMetalArc* is expressed as *is-a-kind* of *ArcWelding*.

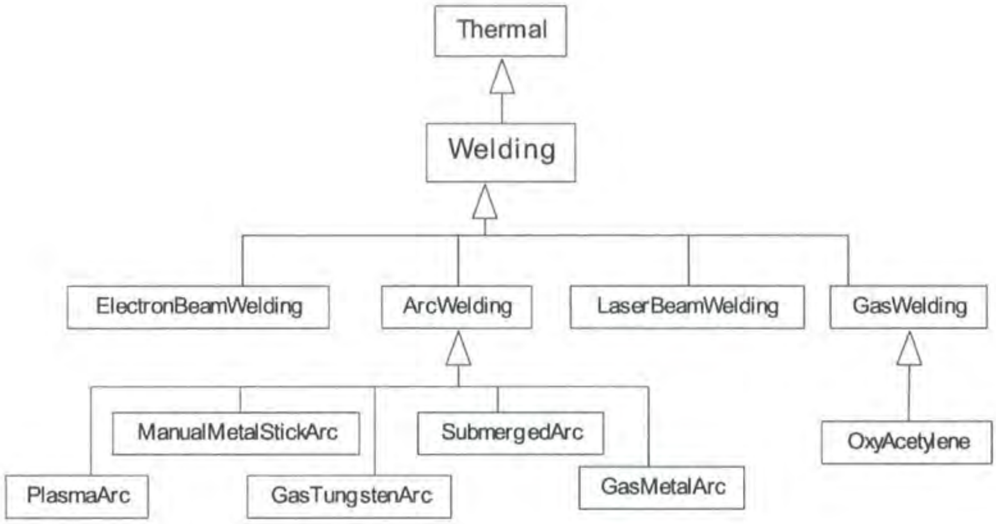


Figure 5.9: Additional Classes for *Thermal* Joining Process

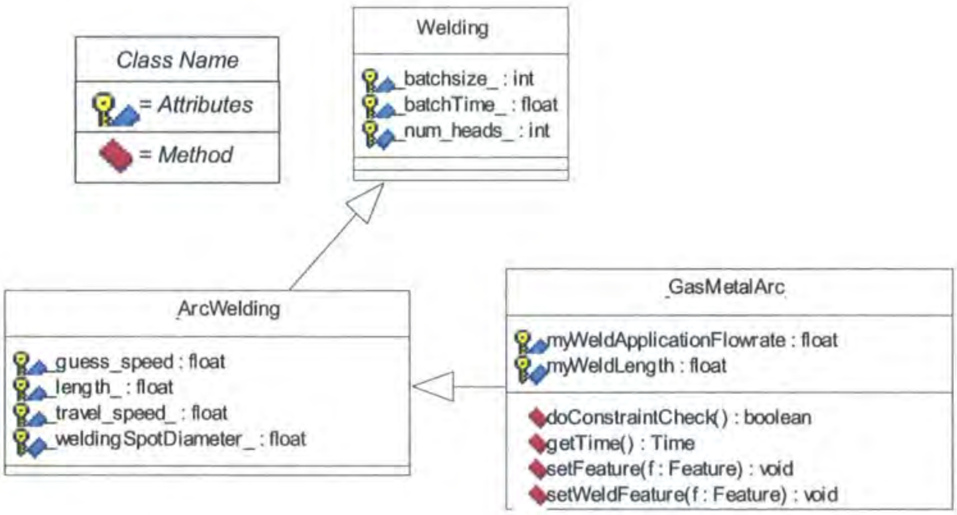


Figure 5.10: Example of Attributes and Methods in *Welding* Class

5.3.3 The Resource Model

According to Maropoulos *et al.* 2003b, the resource model acts a digital representation of an enterprise, suitable for early planning, which captures the capability of factories, machines and humans. Since many manufacturing companies rely on outsourcing operations to supply chain companies, for technical or economic reasons, the resource model is specifically designed with high level classes to model external suppliers. The resource model is composed of two libraries of classes, namely *resource* (the physical entities) and *resource types* (the behaviours). To construct a resource model, the physical resources are first instantiated and then *resource types*, describing the process-related capabilities of each resource, are associated with the resource. This object-oriented structure provides the ability to represent hierarchies of resources at different levels of abstraction as occurs in the real world. The majority of the functionality of the resource classes is contained within the top level resource object as in Figure 5.11.

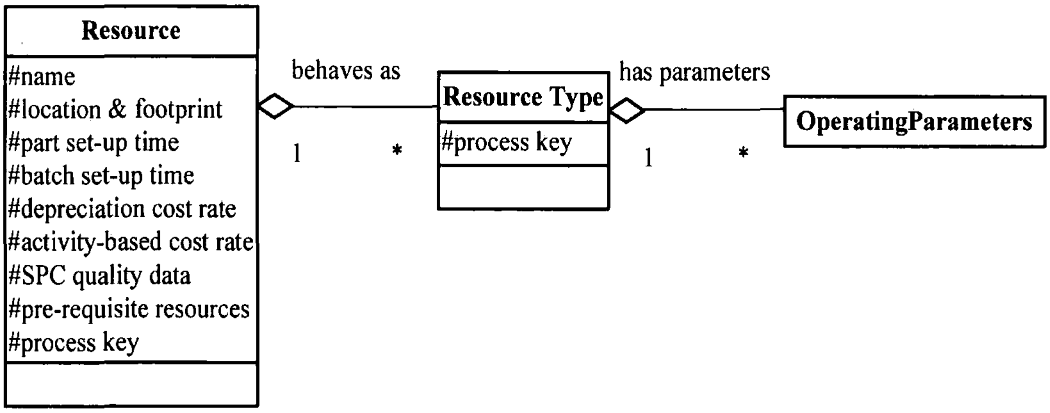


Figure 5.11: Resource and Resource Type Classes with their Attributes

5.3.3.1 The Resource Class

The *resource* class is intended to model the constraints involved in the real world i.e. the factory resource entities. Therefore, when creating resource objects of the *resource* class, the following types of information must be entered of which the factory model is embedded and directly linked and delivered the following points Maropoulos *et al.* 2003b:

1. Footprint

The *footprint* describes the physical position and area taken up by the resource, referenced from the footprint of the parent resource object. For example, a machine will be located within a cell, which in turn is located in a higher-level factory resource. This allows a 2D graphic visualization of the layout of a factory resource. Functions exist to calculate the distance from one resource to another by taking advantage of the hierarchical nature of the resource model.

2. Process Key

The *process key* attribute of a resource identifies the processes which may be executed on it. When a resource type is added to a resource, the type's process keys are appended to the resource.

3. Pre-requisite Resources

Resources, which are dependent upon other resources, such as a machine which requires labour to operate it, also require a description of all the *pre-requisite resources*.

4. Quality

Each resource maintains a historical record of the measured manufactured features. At the point of quality assessment within the process planning algorithm, this record is searched to find the best possible matching criteria in terms of the feature, its dimensions and process parameters.

5. Cost

The cost model at this stage is not at the detailed design stage, it is only used to perform calculation in the activity-based cost rate using direct labour and machine time associated with a particular resource.

6. Delivery

Time information is split into five categories namely: cycle, part setup, batch setup, transport and lead.

- Cycle time is derived directly from the process model based upon resource operating parameters such as maximum power and tool speed, and product characteristics such as dimensions and materials.
- The Part Setup time is user defined for a resource and represents the time required to setup this resource for a single part.
- The Batch Setup time which is also user defined, represents the time required to perform a setup on the resource for a batch of parts.
- Lead Time is again user defined though is only for bought-in parts and reflects the purchasing lead time.
- Transport Time is directly dependent upon the resource model as it utilises the foot-print and transport objects for the calculation of transport time.

7. Knowledge Statements

Statements of knowledge can be attached to objects of the product, process and resource models via a XML Parser (see Sections 4.3.3.1 and 7.3.2) to provide "Knowledge-enriched process plans".

5.3.3.2 The Resource Types

The *resource type* classes describe the processing capability of a resource through a series of *operating parameter* classes. A resource may have more than one resource type, for example a lathe can operate as a turning centre or a drill. The hierarchy of *resource type* objects used in CAPABLE System, is given in Figure 5.12. Each resource type contains a list of *process keys* and a list of *operating parameters.* A *process key* is a textual identifier for a corresponding class of the process model. When a process type is added to a resource, the type's process keys are appended to those of the resource. The process planning then queries the resource process keys to identify potential resources for a given process requirement. Table 5.1 shows a set of *operating parameters* for the resources ranging from the degrees of freedom and tool speed to feed rates. The operating parameters are a critical set of variables that describe processing capability in an aggregate manner. The required operating parameters are obtained from the simplification of detailed process models and their values are derived from literature, the simplification of process optimisation algorithms and databases of production equipment manufacturers. These were identified from a search of the databases and from the necessary algorithmic process models required for obtaining accurate QCD (Quality, Cost and Delivery) values (McKay *et al.* 2001a and b).

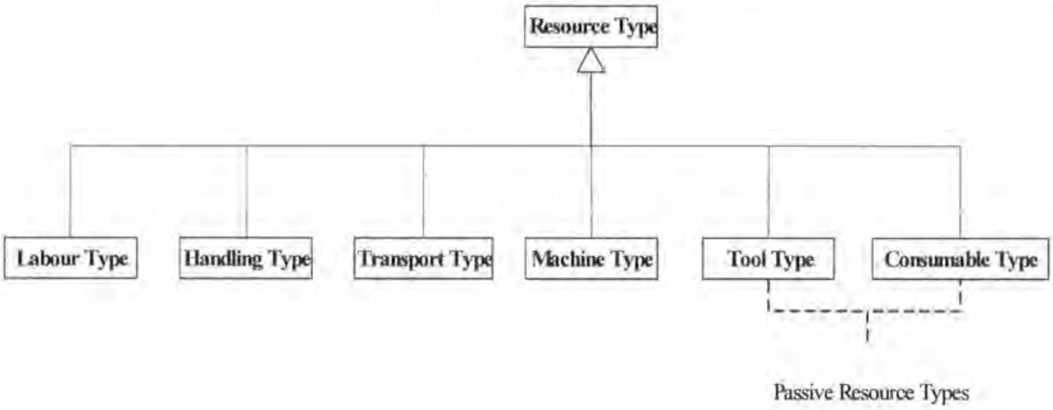


Figure 5.12: Hierarchy of Resource Type Objects

WP_GasMetalArc_MIG		
Operating Parameter	Default	Unit
Max_Travel_Speed	0.25	m/min
Max_Robot_Arm_Length	0.5	m
Max_Weld_Chord_Diameter	0.014	m
Max_Weld_Flowrate	0.05	m^3/min
Max_Weld_Length	0.07	m
Max_Batch_Time	60	s
Max_Batch_Size	1	-

Robot_Handling		
Operating Parameter	Default	Unit
Max_Feeding_Velocity	1.2	m/min
Max_Travel_Distance	0.2	m

MC_Cutting_Mechanical_Drill		
Operating Parameter	Default	Unit
Max_Axial_Feedrate	2	m/min
Max_Axial_DOC	0.5	m
Max_Tool_Speed	3000	rpm
Max_Tool_Radius	0.04	m
Tool_Approach_Dir	z-dir	

MC_Cutting_Mechanical_Mill		
Operating Parameter	Default	Unit
Max_Z_Feedrate	2	m/min
Max_Z_DOC	0.4	m
Max_Y_Feedrate	1.5	m/min
Max_Tool_Speed	4000	rpm
Max_Tool_Radius	0.1	m

Table 5.1: Example Operating Parameters for Various Processes in Welding and Machining

5.3.3.3 Development of the Resource Type Classes

The enhancements to the resource type classes are shown in Figure 5.13 and 5.14 respectively. Figure 5.13 illustrates the resource types for *Mabey & Johnson Ltd*. The additional sub-classes are focused on *Machine_Type* and *Tool_Type*. *MC_Welder* is-a-kind of *Machine_Type* which consists of *LaserWelder, ArcWelder, GasWelder and FlameWelder*. The *MC_Welders* and its associate sub-classes are mainly used to model and capture the parameters such as sizes, speed, length and temperatures of welding. In Mabey & Johnson Ltd, the production process of welding mainly uses *RoboticArcWelder and ManualMetalArcWelder*, both are a-kind-of *ArcWelder* as illustrated in Figure 5.13. In the cutting process category, there are *LaserCutter, ArcCutter, FlameCutter, and MechanicalCutter.* However cutting processes are rarely used within the factory as the dimensions of the beams were already customized before they arrive at the plant.

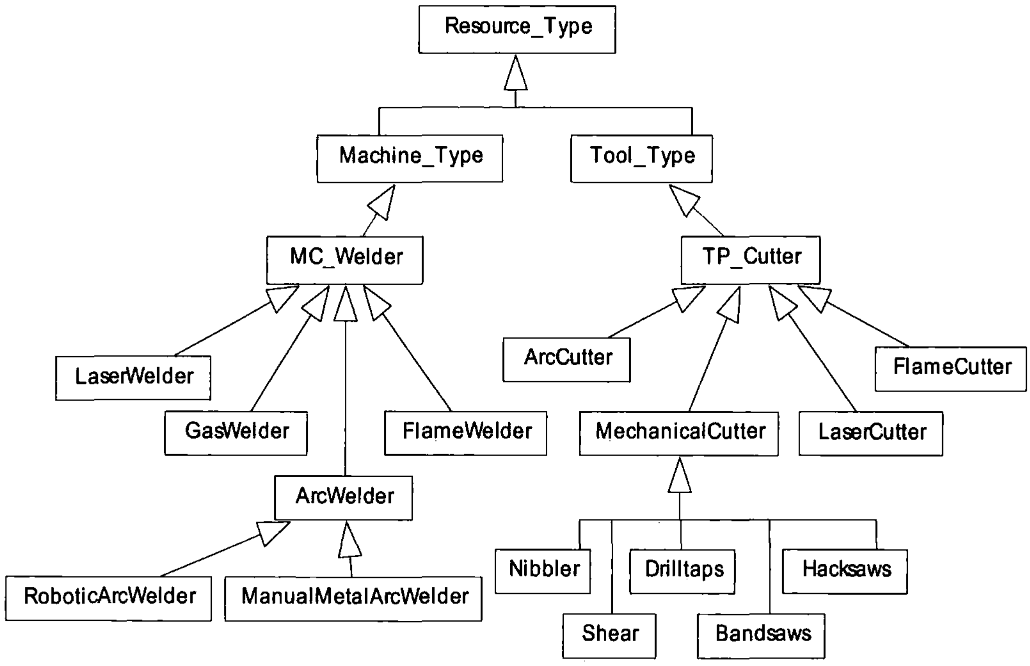


Figure 5.13: Additional Classes for *Machine_Type* and *Tool_Type* (*Mabey & Johnson*)

Figure 5.14 shows further additional resources that have been implemented to model and accommodate the need for *ArvinMeritor*'s vehicle-door-latch assembly sequences. The additional classes are focused on jig tools which consist of different types of feeders and escapement mechanisms (*Auto_Tapping_Machine* and *Turner_Riveting_Machine*) and positioning tools (*Probe* and *RDP_Transducer*).

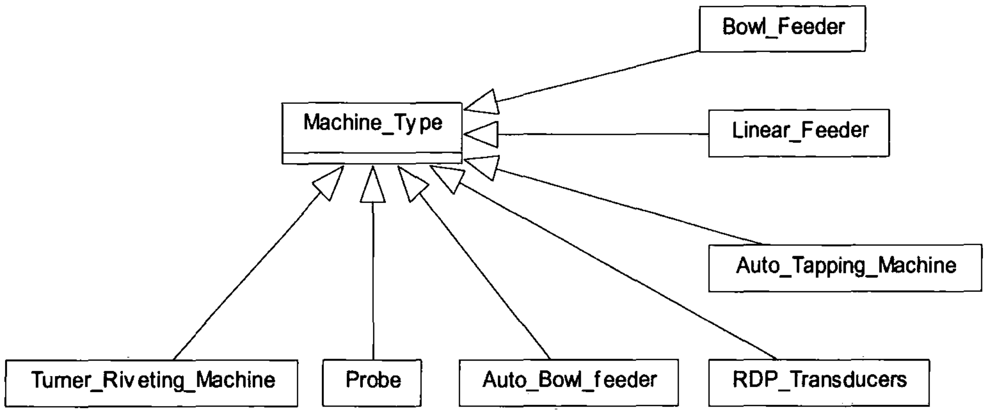


Figure 5.14: Additional Classes for Machine_Type (*ArvinMeritor Automotive*)

5.4 FACTORY MODELLING

The modelling of a factory depends on the resource model in the CAPABLE system which is composed of two sets of classes, namely the *resource* and *resource type* classes as described previously in Section 5.3.3. The resource modelling functionality has been implemented in a truly distributed manner by utilizing the Java remote method invocation (RMI) (Maropoulos *et al.* 2003a; Siple 1988), in which the methods of remote Java objects can be invoked from other Java virtual machines, on different hosts. Access to the database is controlled via a series of private and public areas. The intention is to allow various users to share the core models securely across the distributed enterprise. The concept of aggregate planning demands that the majority of data required for the main constituent models should be available early in the design cycle. Thus, a simple and robust user interface has been constructed to facilitate the distributed and rapid generation of resource models using Internet-based tools, with little or no training required. This means that supply chain companies can also easily model their own factories and use these models for the evaluation of producing parts and subassemblies required by a production network.

5.4.1 Example of Factory Modelling

An annotated screenshot of the user interface of the factory modelling application is shown in Fig. 5.15. The illustration shows the representation of the factory layout of Mabey & Johnson Ltd. On the left of the illustration it is indicated the names of the cells. Within every individual cell, it contains a number of stations used to perform the manufacturing operations. For example, in Galvanising Trailer cell, it contains three different stations, namely surface coating, galvanising and immerse washing. Other cells such as Robot_Cell_0, Robot_Cell_01 and so on contain different welding stations. The capability of every station has been coded within a Java Programming environment. Figure 5.16 depicts how the factory modeller can be used in association with the Product Model to generate the process plan using the "Resource-Aware Planning" function of the CAPABLE System (Maropoulos *et al.* 2003b). The key goal of the factory model is to capture all the functional parameters required to specify a company's manufacturing resources and the associated capabilities. With the techniques of "product features to processes" and, "processes to resources" mappings, the system intelligently explores the capabilities of the complete extended enterprise to find the most suitable processes and production resources for realizing a product. The plan that is generated represents a near optimum solution for the complete product.

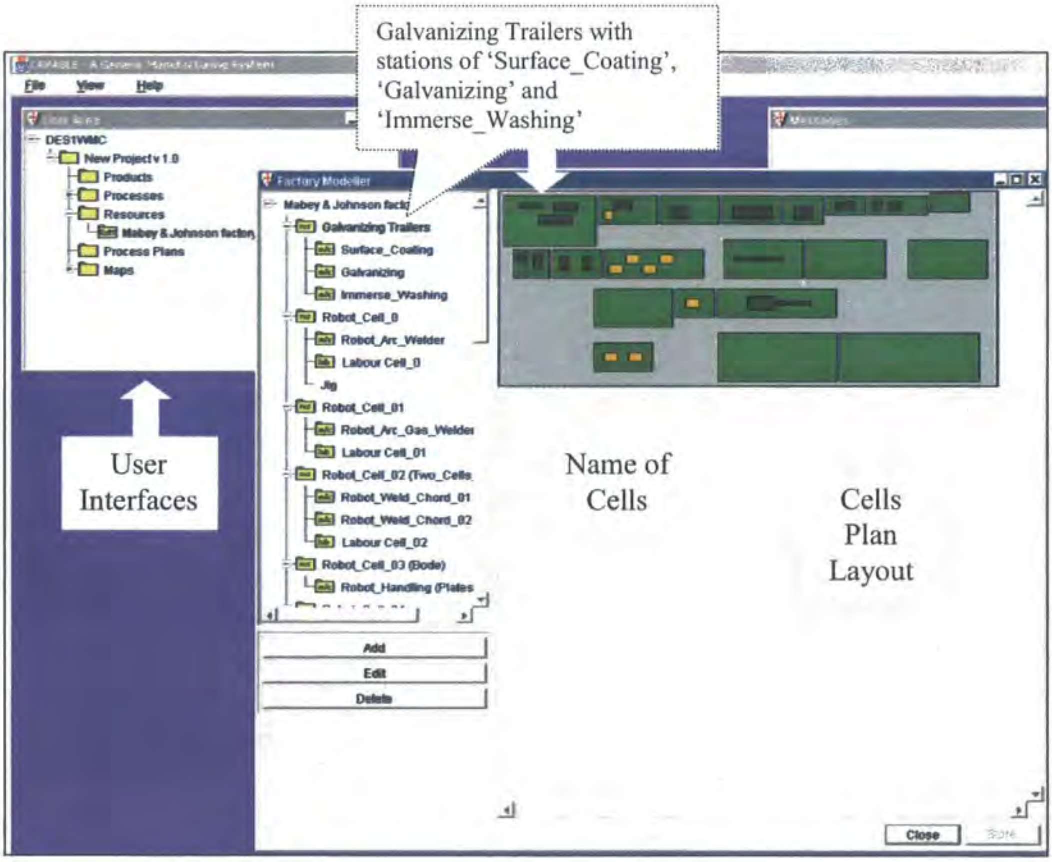


Figure 5.15: Factory Model of Mabey & Johnson Ltd

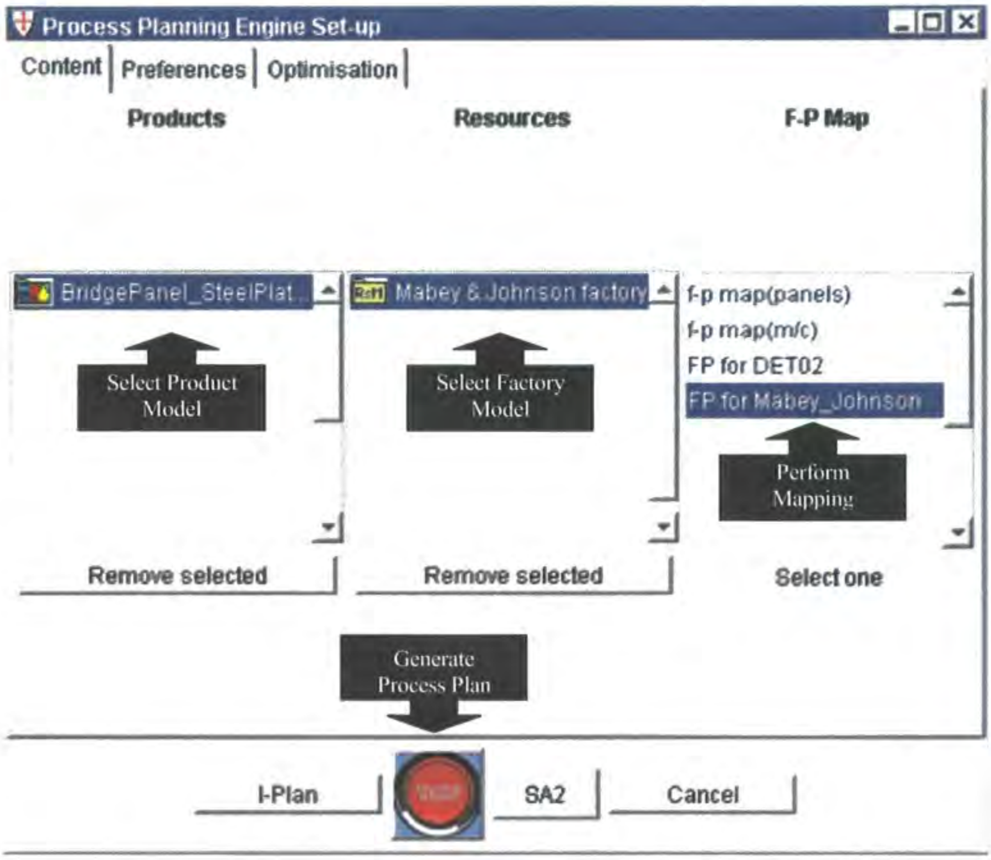


Figure 5.16: Product Model and Factory Resource Mapping

5.5 KNOWLEDGE-ENRICHED AGGREGATE MANUFACTURING MODELS VIA XML PARSER

One of the key developments to enhance the CAPABLE system is to establish an additional function on each of the aggregate manufacturing models. This function is used to invoke the XML Parser to extract XML-formatted knowledge statements via a PDM system. An example of the function for an aggregate resource model (also known as a factory model) is shown in Figure 5.17. The XML Parser is an independent data exchange mechanism. The implementation of the XML Parser is described in Chapter 7.

The primary interest of establishing this link is that a process plan can be generated and improved based upon the knowledge-enriched aggregate manufacturing models. An example of a set of knowledge statements related to a factory model is shown in Figure 5.18. The knowledge statements are the past experience related to product design, manufacturing processes and factory resources which are captured in the knowledge-based system (see Chapter 6). Each of the statements is modelled as a *value* using axioms and constraints (see Chapter 6) which will be prioritised via the Capability Analysis (CA) method (Baker and Maropoulos 1998) to identify potential design or implementation problems with the process plan and feed them back to designers to prompt further detailed analysis or re-design.

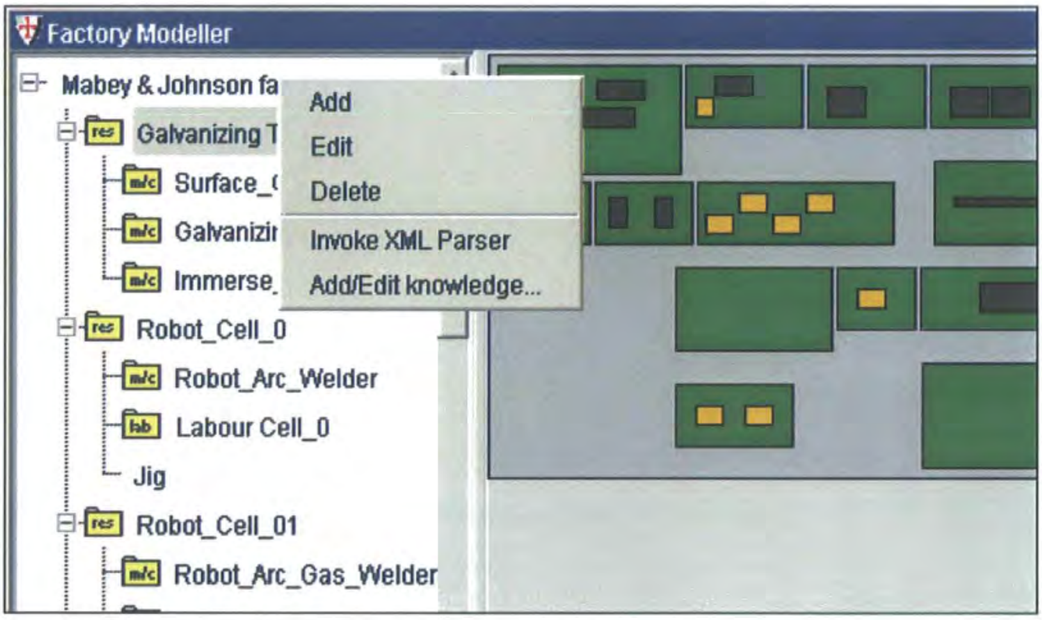


Figure 5.17: The XML Parser function for a Factory Model

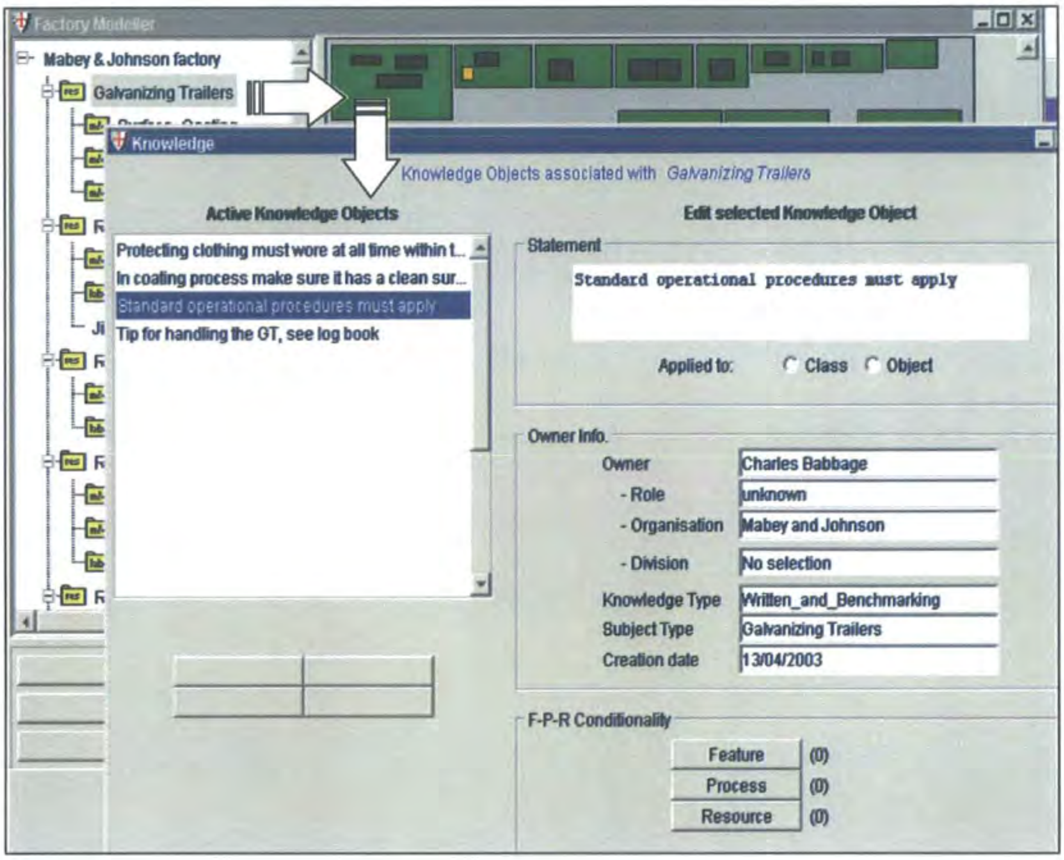


Figure 5.18: Example of Knowledge Statements Related to a Factory Model

5.6 SUMMARY

This chapter has briefly outlined the scope of the aggregate manufacturing models namely product, process and resource of the CAPABLE System. The facilitation and development of the CAPABLE system was the main discussion in this chapter. The additional production capabilities and the type of products have been discussed. In particular the methodology of creating a factory model and the introduction of "weld feature", "joint feature", "orientation" operations and non-shape changing assembly classes (Section 5.3). Factory Models based on the collaborator's production layout have been implemented (Section 5.4). One of the main contributory features of additional development of the CAPABLE system (Section 5.5) is that it enables the system to read and extract XML-based manufacturing knowledge which will be documented in Chapter 8.

The focus of Chapter 6 will describe the implementation and development issues in relation to an organisational knowledge management framework. Both the CAPABLE and Organisational Knowledge systems will contribute a significant impact in bridging the gap of the disconnection in sharing manufacturing knowledge and early process planning evaluations as described in Chapter 1.

Chapter 6

THE ORGANISATIONAL KNOWLEDGE FRAMEWORK

6.1 INTRODUCTION

The organisational knowledge framework is one of the key components in the product development integration architecture as highlighted in Chapter 4. This chapter describes the theoretical aspects and applications of a novel methodology for exploiting a knowledge management editor tool that structures organisational knowledge. The term "organisational knowledge" is defined as "a collective wisdom of a firm which may be explicit, in the form of databases or documents, or tacit, expressed by action" (Rich and Duchessi 2001). The organisational knowledge framework for capturing and representing "manufacturing know-how" has been developed using an ontological approach. An ontological approach can be used to elaborate the organisational knowledge by defining the semantics to capture the meaning of the terms and axioms (to define a set of rules if applicable) to enhance and encapsulate the way of reusing the knowledge-based system in a collaborative manner within a production network. An axiom is a statement that defines or constrains some aspect of the knowledge model and is intended to control or influence the behaviour of the model (Ontoprise 2004). A widely used, proprietary knowledge management editor tool Protégé2000 (Protégé2000 2003) is used to create the axiom-controlled ontology to support the organisational knowledge framework. The delivery mechanism for the application of the captured knowledge makes extensive use of Web-based technologies such as XML and RDF. The captured knowledge is converted into an industry-standard format and then shared within a web-centric Product Data Management (PDM) system. The key benefit of adopting such an approach is the closer integration between the activities taking place during early design. In particular, the effectiveness of decision making is increased (Maropoulos and Gao 2000).

6.2 KNOWLEDGE-BASED SYSTEMS AND ONTOLOGY

6.2.1 Background to Knowledge-Based Systems

Knowledge-based systems may employ any number of techniques for knowledge representation and extraction of the knowledge that is to be re-used. Some of the common approaches are (Turban and Aronson 2000):

- Rule-based systems encapsulate knowledge in the form of structured 'if-then' statements.
- Case-based systems seek out "best practice" solutions to existing problems and adapt them to solve new and similar problems.
- Model-Based Reasoning uses software models to encapsulate knowledge or to emulate real processes.
- Neural Nets are a network of nodes and connections used to encapsulate knowledge, they can "learn" by using examples.
- Fuzzy Logic is used to represent and manipulate knowledge that is incomplete or imprecise.
- Decision Trees encapsulate decision-making knowledge that can be expressed as sets of order decisions.
- Ontologies are working models of entities and interactions of a particular domain of knowledge or practices, such as 'the process planning activity'.

Thus, different knowledge-based systems can be deployed as the technological means for capturing and managing both explicit and tacit knowledge as part of an organisation's knowledge management initiative. However, prior to building knowledge-based systems, the knowledge that pervades the organisation must be identified and modelled using appropriate acquisition, representation and modelling techniques.

6.2.2 Using Ontologies in Knowledge-Based Systems

As discussed in the Chapter 2, ontologies have the potential to improve knowledge capture, organisation, sharing and re-use, and consequently were the obvious choice in this research, to create a knowledge-based system that supports the organisation knowledge framework. Furthermore, using ontologies in the organisational knowledge framework can provide the following advantages (Ciocoiu *et al.* 2001):

- The sharing of knowledge domains across the WWW,
- They do not rely on a set of rule-based techniques, and
- The capabilities of handling complex and disparate information.

However, modelling organisational knowledge is a very complex task, often requiring a combination of different types of ontology construction techniques. To support the organisational knowledge-based system in product development, the following ontological approaches are considered as being important:

- *Top-level/generic/upper-level ontology*, which organizes generic domain independent concepts and relations explicating important semantic distinctions (Sowa 1995).
- *Application ontology*, which consists of the knowledge of a particular application domain (Van Heijst *et al.* 1997).
- *Domain ontology*, which organizes concepts, relations and instances that occur, as well the activities that take place in a domain (Van Heijst *et al.* 1997).

Thus, to construct an ontology-based knowledge system for this research these three ontological approaches have been adopted.

6.2.3 Selection of Ontology Management Software

When developing an ontology, the first task in the research project was to find a suitable ontology software editor capable of managing all the required ontology types. There are more than ninety ontology editing tools available currently (Denny 2004a). According to (Corcho *et al.* 2003; Su and Ilebrekke 2002; Duineveld *et al.* 2000), several basic questions arise relating to ontology tool selection, these are:

- Which tool(s) give/s support to the ontology development process?
- Does the tool support multi-user access?
- How are the ontologies stored (in databases or files)?

- Does the tool have an inference engine?
- What are the inference mechanisms attached to an ontology language?
- What language(s) is available within the tool for implementing the ontology?
- Does the tool support multiple inheritances?
- Is it possible to import / export the ontology in various formats?
- Does the tool provide / support graph view?

Among the ninety-four tools listed by Denny (2004a and b), the most well established ontology editing tools as given in Table 6.1 were selected for comparison.

Whilst all the selected tools may be used to build ontology schemas, standalone or together with instance data, some offer only limited functionality and user support. Some of the editors surveyed are intended for building ontologies for specific domains. These ontology editors may have enhanced support for information standards that are unique to their target domain, but can still be used for general purpose ontology building. For example, Apollo is a genome annotation viewer and editor. The OpenKnoMe Editor is used to support medical reference terminology. Microsoft's Visio for Enterprise Architects uses an object-oriented specification language (Object Role Modelling language) to model information domains.

After considering the tools as compared in Table 6.1, Protégé2000 was selected for use in this research based on the followings:

1. It is an integrated tool for ontology and knowledge-base editing.
2. It is an open-source, Java-based, extensible architecture for the creation of customized knowledge-based tools.
3. It supports plug-ins to enable different inference engines, query engines and visualisations to be run within it.

Protégé2000 supports a large number of data formats and is usable both as a development environment and as an end-user knowledge management tool. Protégé2000 is one of the very few open-source ontology editors designed to be scalable and customisable and has a wide user community. Indeed, one of the major strengths of Protégé2000 is the support that novice users receive from an online discussion group and the advanced technical support that is offered.

Tool	**Modelling Features/Limitations**	**Base Language**	**Import/Export Formats**	**Graph View**	**Multi-user Support**	**Comments**
The software tool for editing ontologies	The representation and logical qualities that can be expressed in the built ontology	The native or primary language used to encode the ontology	Other languages the built ontology can be serialized in	The extent to which the built ontology can be created, debugged, edited and/or compared directly in graphic form	Features that allow and facilitate concurrent development of the built ontology	Pertinent information about methodology, availability and support, additional features, etc.
Apollo	Classes with slots plus relations; functions; hierarchical views.	OKBC model	CLOS; OCML ; RDF/XML/OWL	Pegasys (built in)	No	None
OpenKnoMe	Description logic terminological modelling without support for individuals or type system.	GRAIL.	CLIPS; XML	No	No	User roles and read/write privileges; version control. Users see each other's changes only when they check modules back in.
Protégé-2000	Multiple inheritance concept and relation hierarchies (but single class for instance); meta-classes; instances specification support; constraint axioms ala Prolog, F-Logic, OIL and general axiom language (PAL) via plug-ins.	OKBC model	RDF(S); XML Schema; RDB schema via Data Genie plug-in; DAML+OIL; XMI-backend for UML; OWL;	Browsing classes & global properties via GraphViz plug-in; nested graph views with editing via Jambalaya plug-in.	Yes	Support for CommonKADS methodology.
Visio for Enterprise Architects	Most object-role modelling (ORM) constructs, but imposes relational logical constraints on specification.	ORM	XML (via add-on); DDL	ORM class diagrams	Yes	ORM modeller may be effective for specifying domain ontologies; part of Visual Studio.NET Enterprise Architect

Table 6.1: Example ontology editing tools. Adapted from Denny, (2004a).

6.3 THE ORGANISATIONAL KNOWLEDGE FRAMEWORK

Figure 6.1, depicts a novel organisational knowledge framework which is an ontology-based environment that has been developed for this research work. The framework is used specifically to create, manage and capture qualitative and quantitative knowledge statements related to manufacturing and assembly processes for complex designs. These knowledge statements exist within a distributed design team as shown. The data exchange format used within the framework is the industry standard XML. A Java-based XML Parser mechanism has been implemented for extracting the knowledge to be reused by a process planning system. The resulting process plans, which contain an evaluation of the likely quality, cost and delivery performance can be stored into an information management system as illustrated. The implementation of this framework is flexible, as the specific information systems can be adapted to the needs of the enterprise. For example, knowledge may be distributed via a centralised PDM system or via a decentralised P2P network (Penserini *et al.* 2003). This has been combined with a system that applies Web-based technologies for the capture of knowledge statements. The captured knowledge is converted into an XML-formatted file and shared within a web-centric PDM system, to support the collaborative product development process.

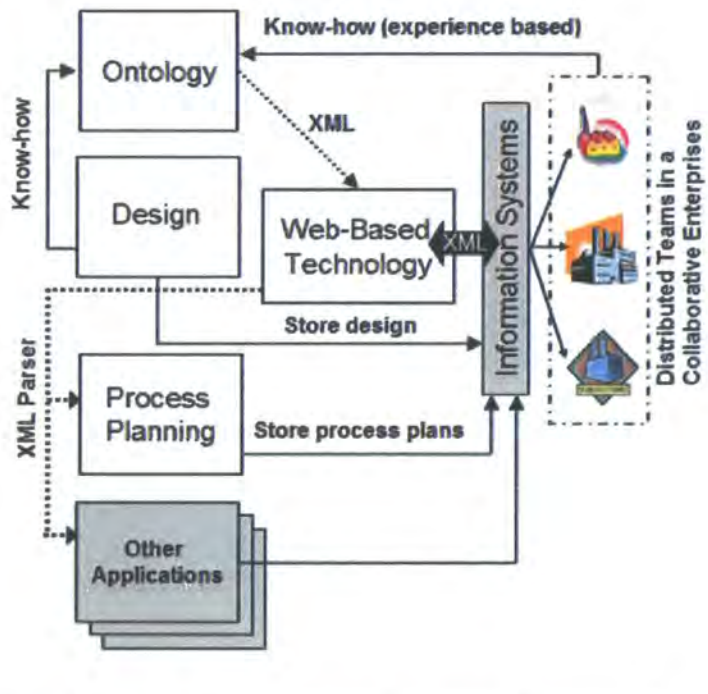


Figure 6.1: The Organisational Knowledge Framework

6.3.1 The General Structure of Organisational Knowledge

There are four kinds of knowledge that are generally recognised as being important in a knowledge-based economy (ITAG 1999).

1. The first, *'Know-what'* is knowledge about facts.
2. The second is *'Know-why'* and refers to scientific knowledge and understanding, for instance, the principles of why things happen. This also encompasses the skills often found in research laboratories or generated as a result of collaborative research between organisations.
3. The third type of knowledge is *'Know-how'* which refers to skills and capabilities, for example, the ability to use a particular machine or skills gained through practice and experience. In industry, '*know-how*' is often used interchangeably with the term knowledge management from design to manufacturing, whether its context is described explicitly or implicitly.
4. The final component is *'Know-who'*, which describes where in the enterprise knowledge is stored. Capturing *'Know-who'* requires a deep understanding of the expertise within an organisation.

The approach to define and construct the organisational knowledge ontology was based on the four kinds of knowledge. In this particular application *'Know-who'* is only used to record the name of the knowledge owner, it has been implemented as an attribute of the *'Know-how'* class. In future versions of the work it may be possible for example to make a separate module, so that meta-information about confidence in the judgement of a person may be recorded.

6.3.2 The Main Organisational Knowledge Ontology

Figure 6.2 shows a class taxonomy of the organisational knowledge ontology which was specifically constructed to model manufacturing knowledge for this research. According to Jenz & Partner (2003), "an ontology is based on a taxonomy which represents a class hierarchy in the object-oriented world". The organisational knowledge ontology consists of three major modules, namely; Organisation Knowledge *'Know-how'*, *'Know-what'* and *'Know-why'*, which are defined as *is_a_kind_of* organisational knowledge. The modules have *constraints* imposed upon them, namely *Probability*, *LargerTheBetter*, *SmallerTheBetter*, *NominalTheBest*, *FactorTarget* and *FactorBenchmark*. However, the research work at this stage is mainly focused on applying *Probability* to define a constraint related to the instances captured within the ontology as this will be discussed in Section 6.3.5 and 6.4.3.

Organisation Knowledge *Know-how* is further broken down into:

1. *'Business_Process_Know_how'*,
2. *'Design_Know_how'* and
3. *'Manufacturing_Know_how'*

The *'Business_Process_Know_how'* module defines marketing strategies, sales, purchasing, vendors and supply chains, suppliers and costing data. *'Design_Know_how'* represents the information on product design and standards, and customisation such as bespoke customer designs. *Manufacturing_Know_how* forms an integral part of the framework and consists of quantitative and qualitative knowledge statements related to the production processes and equipment. It captures production skills, process best practice and experience-based information. The *'Organisation_Know_what'* module is used to define empirical knowledge based upon facts and hypotheses. Finally, *'Organisation_Know_why'* defines principles of why things happen.

The research looked at exploiting the possibility of applying this information to support process planning using the CAPABLE Aggregate Process Planning System. In order to optimise a manufacturing or assembly process plan, *know-how* is required to determine the most suitable processes and equipment to use. Also, by using manufacturing *know-how* and design *know-how* together with the capability analysis technique, the generated process plan can be further improved (Bramall *et al.* 2003, Baker and Maropoulos 1998). Furthermore, once the plan is stored in the PDM system it can be distributed to various locations and shared amongst the collaborators subject to predefined agreements to protect intellectual property and commercial advantage.

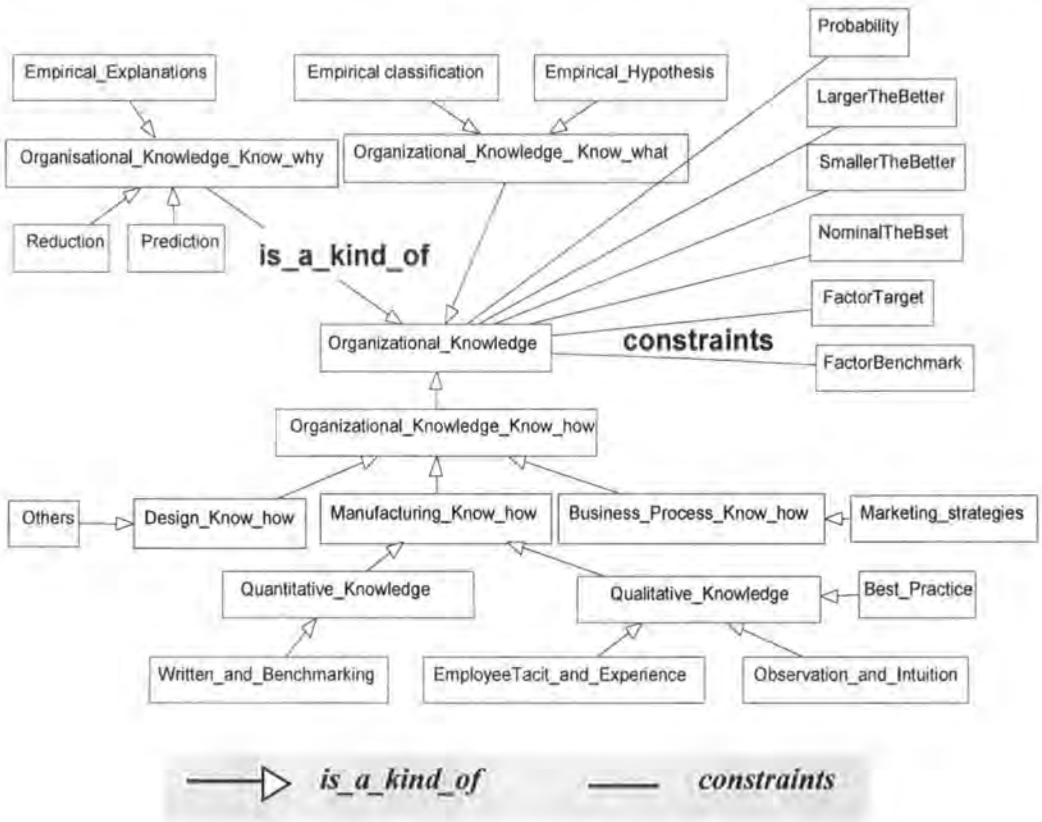


Figure 6.2: A High Level Representation of the Organisational Knowledge Ontology

6.3.3 Constraints of the Manufacturing Know-how Ontology

Currently, a prototype research implementation of the ontology covers the *'Know-how'* category. The description in this section is, therefore, focused on the specification of the manufacturing *know-how* ontology and its implementation. Two knowledge bases relating to design and manufacturing have been developed concurrently. The design KBS (Aziz *et al.* 2004) is used to capture and define issues related to product design rules, geometric representation and design standards such as AP224 (SCRA 2003). The second KBS deals with manufacturing *know-how* related issues. In a manufacturing context, particularly in the early definition of process plans, knowledge representation can be either qualitative or quantitative (Bramall, 2006). In this research, both the qualitative and quantitative knowledge is encapsulated in the ontology as shown in Figure 6.2. *Organizational_Knowledge_Know_how* is the top level of the ontology, which has a series of sub-levels such as *Manufacturing_Know_how*, *Business_Process_Know_how* and *Design_Know_how*.

Manufacturing_Know_how has two further sub-levels, *Qualitative_Knowledge* and *Quantitative_Knowledge* as illustrated in Figure 6.2 and *Qualitative_Knowledge* consists of several sub-classes such as; *Observation_and_Intuition* which is defined as knowledge gained by observing activities and well-understood ideas. *EmployeeTacit_and_Experience* which is defined as knowledge which employees have gained through experience. Lastly, *BestPractice* defines ideas about what works best in a given situation, examples of how not to do something, lessons from a specific program or project in an application. The *Quantitative_Knowledge* class captures the information of *Written_and_Benchmarking.*

Figure 6.3 illustrates a detail representation of the Organisational Knowledge ontology. For example *Organizational_Knowledge_Know_how* consists of a number of object properties namely *Type_of_Knowhow, Owner_of_Knowhow, Date_Prepared* and *Department* which are captured in *string* type statements. Further object properties within this *class* are *Internal_or_External* to denote whether the knowledge is owned by internal or external staff, and *PPR* (*P*roductFeatures, *P*rocesses and *R*esources) which describes the kind of knowledge it belongs to, both of them are represented by *symbol* type statements. All the object properties are inherited by the subclasses further down the structure. One important aspect in designing an ontology is that, object properties can only be inherited down stream within the structure. The subclasses of *Manufacturing_Know_how* and *Design_Know_how* consist of the *Type_of_Manufacturing_Knowhow* and *Type_of_Design_Knowhow* respectively. Both of the subclasses are inferenced by an object property, *ProbabilityFactor*, which represents the level of confidence of the captured knowledge numerically (Bramall, 2006) and (Li *et al*, 2006). *Quantitative_Knowledge* and *Qualitative_Knowledge* are a-kind-of *Manufacturing_Know_how* in this ontology which consist of the descriptions of the knowledge statements and the *KS_type* (knowledge statement type). The *Quantitative_Knowledge* module has also been imposed with *constraints* namely *LargerTheBetter*, *SmallerTheBetter* and *NominalTheBest*. The research work at this stage is mainly focused on applying *Probability* to define a constraint related to the instances. In addition, Figure 6.4 illustrates the actual implementation within the Protégé2000 editor. The illustration shows the structure of the ontology and the object properties of example classes. The tree on the left represents a class hierarchy. The forms on the right show the definition of the *BestPractice* and *Written_and_Benchmarking* classes.

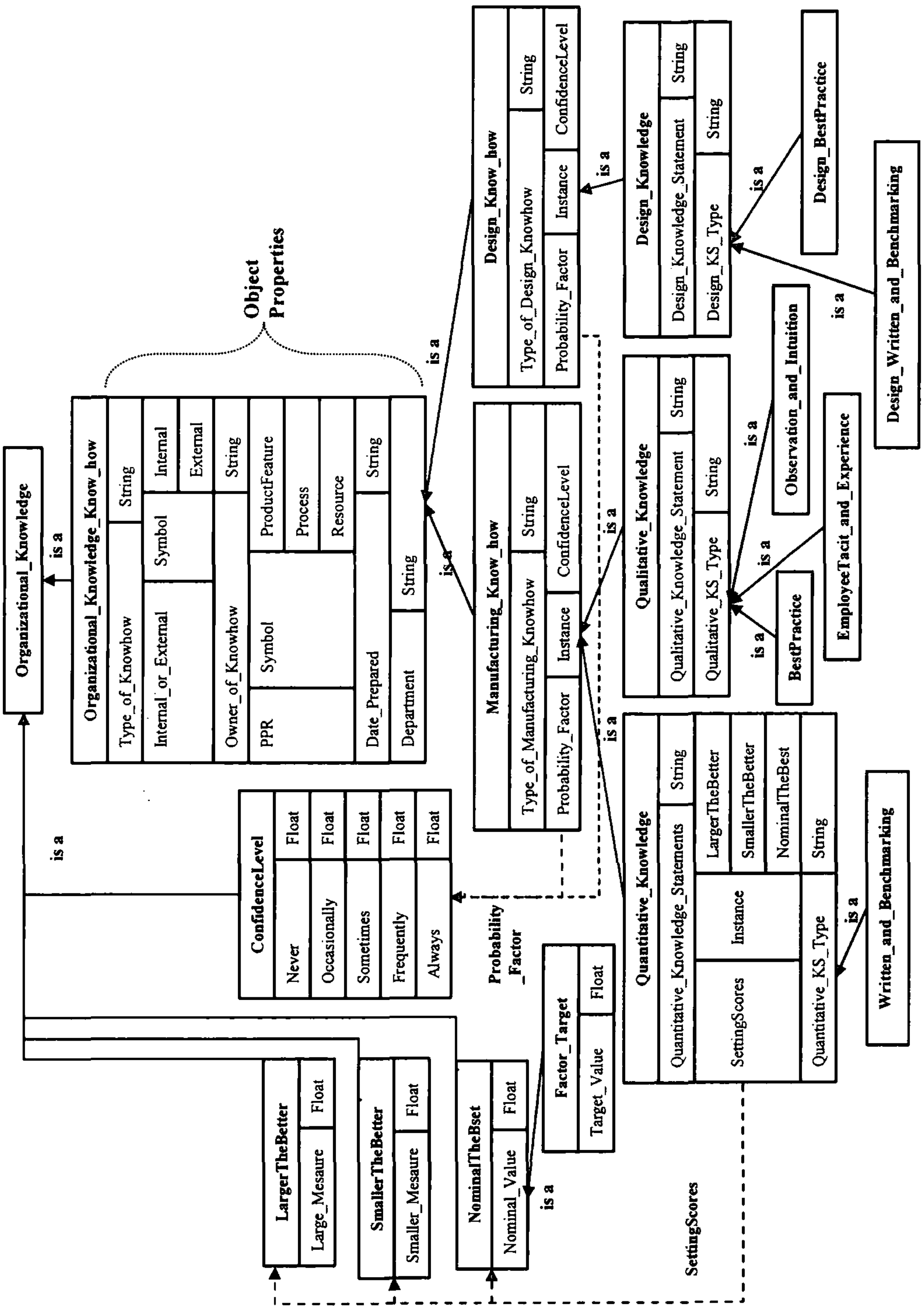


Figure 6.3: A Detail Representation of the Organisational 'Know-how' Ontology

Figure 6.4: Example Snapshots of the Ontology Representing Organisational Knowledge.

6.3.4 An Example of Design and Manufacturing Know-how

To illustrate the application of design and manufacturing *'Know-how'*, a simple example is used, where necessary as a guide through the rest of this chapter. The example is used to demonstrate the theoretical and implementation aspects of the ontological approach in creating the knowledge-based system. The product in the example is a '*light vehicle aftermarket accessory*'', a performance exhaust for sports cars, which consists of two sub-parts A (*exhaust pipe*) and B (*performance muffler*) as illustrated in Figure 6.5.

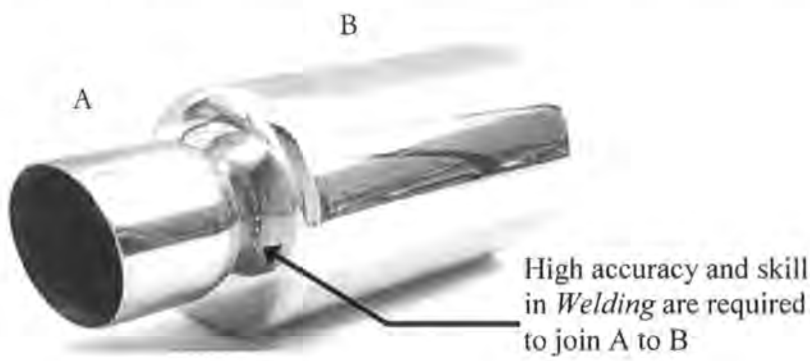


Figure 6.5: Performance Exhaust

The performance exhaust is an expensive high quality accessory and in order to design and assemble part A and B, a high degree of '*Know-how*' in design and welding are needed. As described previously, the manufacturing '*Know-how*' ontology is to be used to capture knowledge of various '*Know-how*'. In relation to the design and welding process, the knowledge types of *Written_and_Benchmarking* and *BestPractice* are considered as vital to lead to the production of a high quality performance exhaust. Typically, the kind of knowledge statements in *Written_and_Benchmarking* and *BestPractice* are shown in Table 6.2 and 6.3, each of the statements has a *Probability Factor* assigned, a value used to determine the experts' preferences. *Probability Factor* represents the level of confidence of the captured knowledge numerically (Bramall 2006) and (Li *et al*, 2006). The levels of confidence are represented by 0 to 1, a value used to determine the experts' preferences.

<table>
<tr><td></td><td colspan="2">Design Knowledge (Know-how)</td></tr>
<tr><td rowspan="3">Written and Benchmarking</td><td>Knowledge Statement</td><td>Probability Factor</td></tr>
<tr><td>The benchmarking of good design for filling, venting and drainage requires:-
▪ Means for the access and drainage of molten zinc
Means for escape of gases from internal compartments (venting)</td><td>0.75</td></tr>
<tr><td>Design of holes for venting and draining should be as large as possible. The minimum hole diameters are given in the following table:
<table>
<tr><td>Size of hollow section (mm)</td><td>Minimum diameter of hole (mm)</td></tr>
<tr><td><25</td><td>10</td></tr>
<tr><td>≥25-50</td><td>12</td></tr>
<tr><td>>50-100</td><td>16</td></tr>
<tr><td>>100-150</td><td>20</td></tr>
<tr><td>>150</td><td>Consult galvanizer</td></tr>
</table></td><td>1.0</td></tr>
<tr><td rowspan="2">Best Practice</td><td>In designing components which require galvanising, overlap surfaces should be avoided as far as possible. i.e. if overlaps are completely sealed by welding there is a risk of explosion during dipping due to increased pressure of any entrapped air.</td><td>0.75</td></tr>
<tr><td>Efforts can be made at the design stage to minimise residual stresses:-
▪ Controlling welding procedures during fabrication
▪ Arranging weld seams symmetrically. The size of weld seams should be kept to minimum.</td><td>1.0</td></tr>
</table>

Table 6.2: Typical Example of Design 'Know-how'

	Manufacturing Knowledge (Know-how)	
	Knowledge Statement	*Probability Factor*
Written and Benchmarking	The bottom line; when welding parts < 2 mm its imperative the plant uses a 035 or 040 wire.	0.75
	The bottom line; when welding parts > 2 mm its imperative the plant uses a wire above 040.	1.0
Best Practice	Do not weld on coated metals, such as galvanized, lead, or cadmium plated steel, unless the coating is removed from the weld area, the area is well ventilated, and while wearing an air-supplied respirator. The coatings and any metals containing these elements can give off toxic fumes if welded.	0.75
	Welding on closed containers, such as tanks, drums, or pipes, can cause them to blow up. Sparks can fly off from the welding arc. The flying sparks, hot workpiece and hot equipment can cause fires and burns. Accidental contact of electrode to metal objects can cause sparks, explosion, overheating, or fire. Check and be sure the area is safe before doing any welding.	1.0

Table 6.3: Typical Example of Manufacturing 'Know-how'

6.3.5 Method of Measuring Knowledge Statements

Aggregate Process Planning has previously been defined as the task of identifying production requirements for an embodiment-level design to provide the basis of decision support through the technical analysis of manufacturability considerations during early product development (Maropoulos *et al.* 1998). CAPABLE (Bramall *et al.* 2003) is an implementation of an aggregate planning system, which has specifically been developed to assist the integrators of large, complex products by exploring production options. The system decomposes the product design into features and intelligently explores all the possible production alternatives, ultimately allocating the production of specific parts to factories and equipment. CAPABLE is designed to operate during the earliest phases of design where the quantitative analysis of design solutions is not always possible, it therefore uses the concept of 'capability' to measure the likely performance of a plan. The idea of capability is to model an expert's preference for the selection of a particular process or piece of equipment in the process plan, derived from one of the three types of enterprise knowledge. A capability factor is a pre-defined measure of performance, related to a specific area of the plan that the enterprise seeks to improve. Typical examples are shown in Tables 6.2, 6.3 and Appendix F, where each of the knowledge statements has been assigned a value by the domain expert. For each probability factor, past experience captured in the ontology as *know-what, know-how* and *know-why,* is modelled as a value which describes the achievement towards the performance target which would be obtained if the entity was selected in the plan. This may be defined quantitatively or qualitatively using a scale ranging from 'strongly desirable' to 'strongly undesirable'. Hence, capability factors can directly utilize the ontology as a source of knowledge.

A Capability Analysis (CA) method (Baker and Maropoulos 1998; Bramall *et al.* 2002) has previously been developed to prioritise knowledge values associated with product, process and resource entities. When applied to the process planning problem, CA makes it possible to identify potential design or implementation problems with the process plans and feed them back to designers to prompt further detailed analysis or re-design. The specific purpose of CA, carried out within CAPABLE, is to examine the performances attributed (via *know-what, know-how* and *know-why* factors) to the product, process, resource elements of a process plan, and assign a priority to each performance to form a ranked list of factors that require investigation or action at each of those levels.

This section so far has discussed the theoretical aspects of using an ontology in a knowledge-based system. In the remainder of this chapter, the discussion will focus on the implementation to build up such system.

6.4 CONSTRUCTION OF THE ONTOLOGY-BASED KNOWLEDGE SYSTEM

6.4.1 The Activities to Design the Knowledge-Based System

To design a knowledge-based system using the ontology technique is a complex task. The approach in this work proposes a way of constructing a knowledge model which involves four activities as illustrated in Figure 6.6:

1. The first activity in KBS construction is to form a UML representation of the ontology and define the relationships among the classes.
2. Transfer the UML class taxonomy into the application ontology using XMI and subsequently mapping the roles and the concepts in the ontology.
3. Configure and define the ontology. This activity involves the construction of an application-specific ontology. In general, ontology construction is a difficult process that requires the expertise of knowledge engineers and domain experts. A library of reusable ontological theories can ease this process. The knowledge engineers and domain experts work as a team to select the reusable theories and, if necessary, tune them and define an agreeable vocabulary to meet the demands of the application.
4. Instantiate the application ontology with domain knowledge, for example, the knowledge statements as shown in Tables 6.2 and 6.3 are a kind of domain knowledge. While the application ontology defines which concepts are used in the domain, the application knowledge describes the actual instances of these concepts. Hence, this requires the domain experts to gather all necessary knowledge to instantiate the domain ontology.

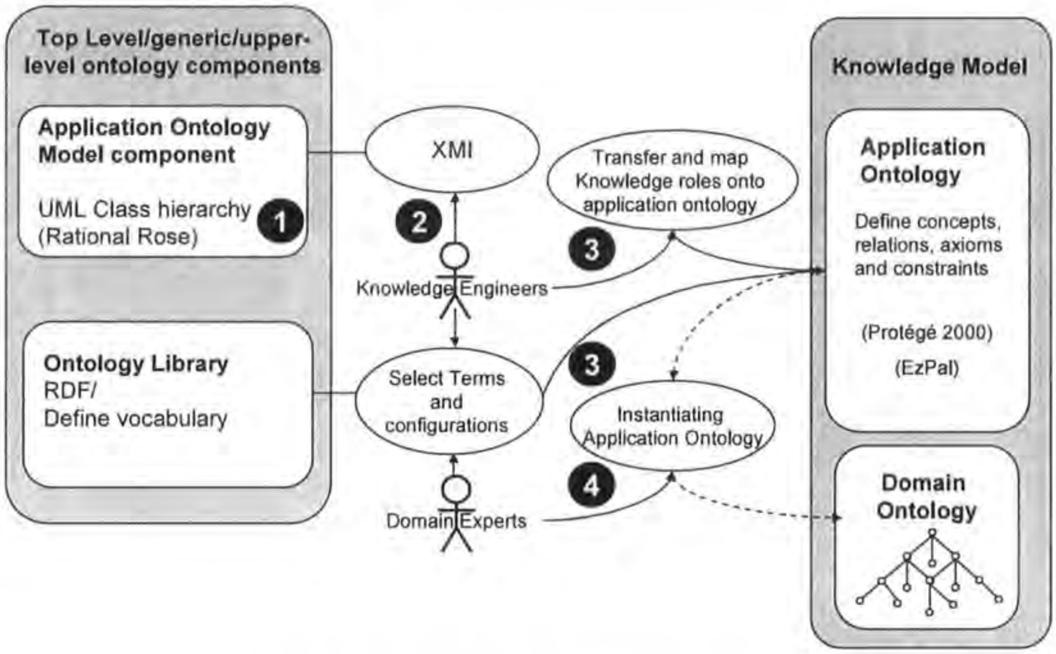


Figure 6.6: Activities in the Design of an Ontology-Based Knowledge System

6.4.2 Implementation of the Knowledge-Based System

The design and manufacturing KBSs have been developed in Protégé2000 for capturing early manufacturing know-how using the ontology. The class taxonomy shown in Figure 6.2 (Section 6.3.2) was initially constructed to model the domain using the Unified Modelling Language (UML) (Appendix G - G1). The UML class diagram was imported into the Protégé2000 KBS Editor, via XMI (XML Metadata Interchange Format) (see Appendix G - G2 and G3) to create the resulting ontology as shown in figures 6.3 and 6.4. XMI is an open industry standard for applying XML to abstract systems such as UML. The intention of XMI is to propose a way to standardize XML for users to exchange information about metadata in distributed heterogeneous environments (OMG, 2005). XML bridges part of the gap by providing the building blocks for "serializing" UML data textually. XMI is required for complex ontology-based systems, such as the one proposed, because it can capture and express the *relationships* that can be expressed using UML class diagrams (Laird 2001).

XMI is available as a standard plug-in for most UML modelling tools such as Rational Rose[1], Ideogramic[2] and ArgoUML [3]. The XMI plug-in allows all different UML products to exchange information in UML class diagram format. Rather than single iterations, round trips are possible so that even reverse-engineered Rational Rose UML class diagrams can flow back to the Protégé2000 KBS Editor, Ideogramic and ArgoUML and serve as the basis for further collaboration and refinement.

For additional refinement of the ontology within the Protégé2000 environment, there are two individual processes which can be applied in combination to form the ontology.

1. use the vocabulary from the RDF library (RDF Vocabulary Description Language 1.0, 2004), or
2. using the knowledge and domain experts' own vocabulary.

In this case the ontology was formalised using the developer's own vocabulary as shown in figures in 6.3 and 6.4. The application ontology will then be instantiated with domain knowledge as the resulting domain ontology, which is also known as the KBS created using the graphical user interface shown in Figure 6.7. As for Figure 6.8, the example illustrates the resulting instances of (a) *Written_and_Benchmarking* and (b) *BestPractice*. Each of the instances has a set of object properties to capture the required information. For example, in relation to the performance exhaust product as described in Section 6.3.4, the knowledge statements and probability factors can be captured within the user dialog box as indicated in Figure 6.7. In addition, Figure 6.7 also shows a *Type_of_Knowhow* instance for 'Manufacturing'. The *Product_Process_Resource (PPR)* slot describes the entity in the manufacturing system which the instance data relates to. Other fields contain specific object property data to describe the knowledge, for example owner and date created. The Design knowledge base was also developed using the same process. One of the advantages of using Protégé2000 is that the information can be converted between formats using a native function, which eliminates the process of manually implementing data exchange mappings which would have been required if the KBS was not XML-enabled. A Java-based XML Parser was developed, as described in Chapter 4, (Section 4.2.2) to

[1] http://www-306.ibm.com/software/rational/
[2] http://www.ideogramic.com
[3] http://argouml.tigris.org

instantiate Java objects directly from the XML file and associate them with the process and equipment models in the planning system. Hence, this allowed the design and manufacturing KBSs to be shared and re-used by the capability analysis methods of the CAPABLE System and other third party software.

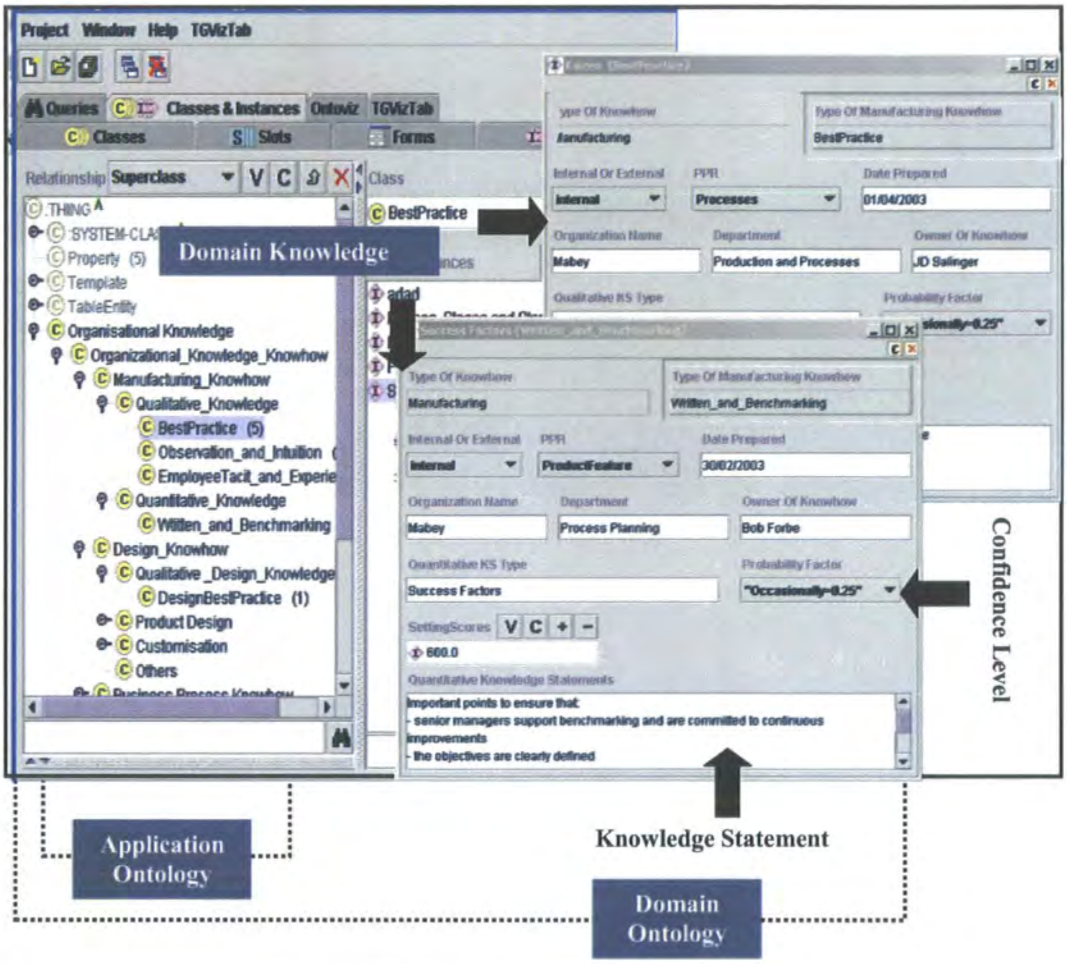
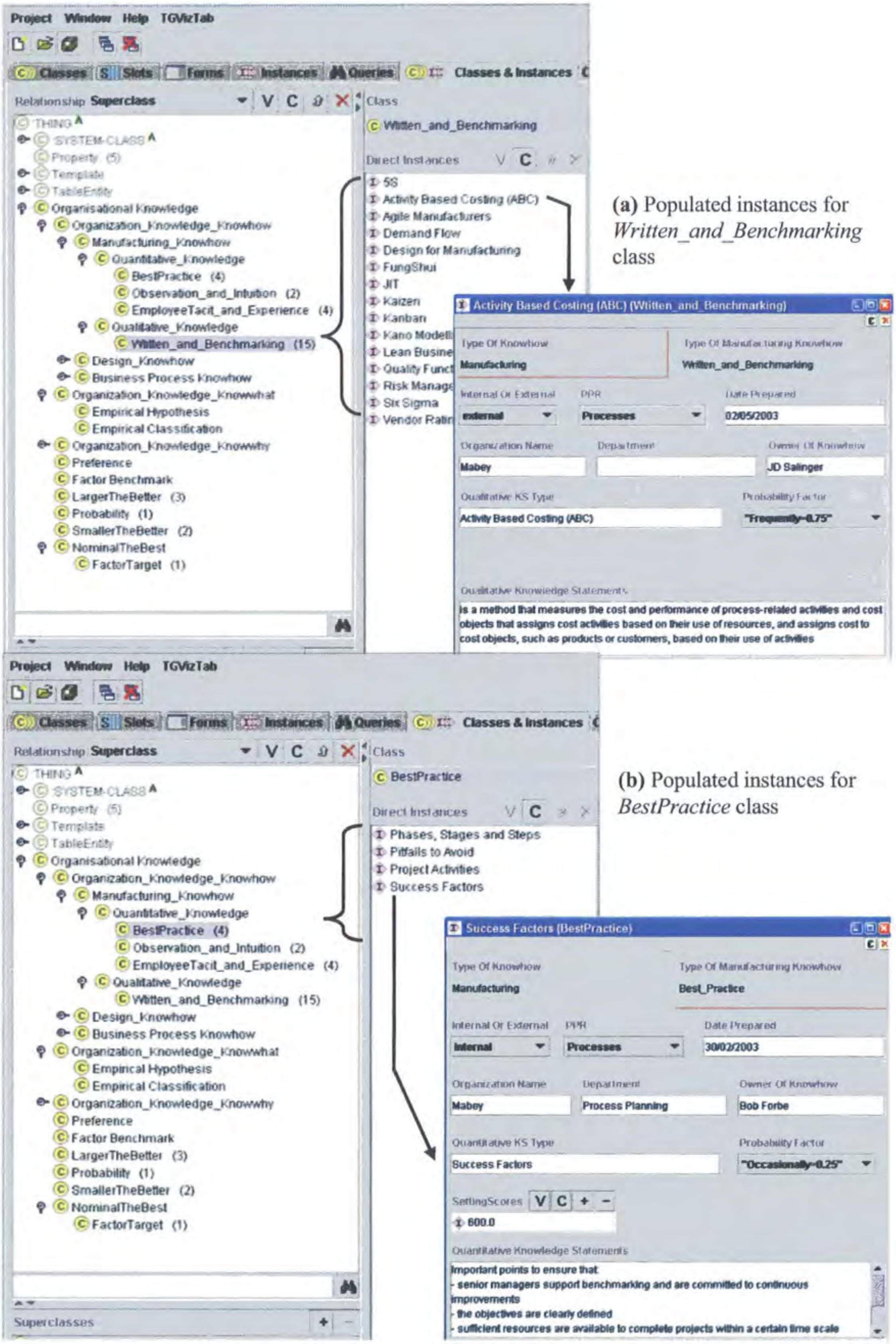


Figure 6.7: The Knowledge-Based System



(a) Populated instances for *Written_and_Benchmarking* class

(b) Populated instances for *BestPractice* class

Figure 6.8: Example of Instances

6.4.3 The Application of Axioms and Constraints within the Ontology

It is important to be able to define axioms and constraints within an ontology. The axioms and constraints are used to define a specific value or condition in relation to a specific knowledge statement. For example, the design and manufacturing 'Know-how' for the performance exhaust as described in Section 6.3.5, where the tables of the knowledge statements have been assigned different values by the domain expert. The methodology of constructing these values in the knowledge-based system will be defined using axioms and constraints as described in the following.

In terms of implementation of axioms and constraints within the ontology, there are several plug-ins available within the Protégé2000. The EZPal tab (Hou *et al.* 2002) is designed to facilitate the acquisition of Protégé Axiom Language (PAL) based constraints without the need to understand the language itself. The plug-in uses a library of templates based on reusable patterns of previously encoded axioms. The interface allows users to compose constraints using a 'fill-in-the-blanks' approach. The EZPal tab makes use of a Protégé2000 ontology to store three major categories of information which is classified as *Property, Template* and *Pattern* as shown in Figure 6.9.

- A *property* is an abstract description for the common features of a group of *templates*. Properties are not mutually exclusive: each *template* may satisfy more than one property.
- Each *template* describes a set of frequently used axiom design *patterns* based on their semantic and structural similarities. It stores the relevant 'variation' information to allow retrieval of a specific *pattern* to allow value entries for axiom generations.
- A *pattern* is defined as a logical sentence derived from a group of axioms that are structurally identical except for specific references. Individual *patterns* are not stored explicitly in the library but further generalized into *templates*.

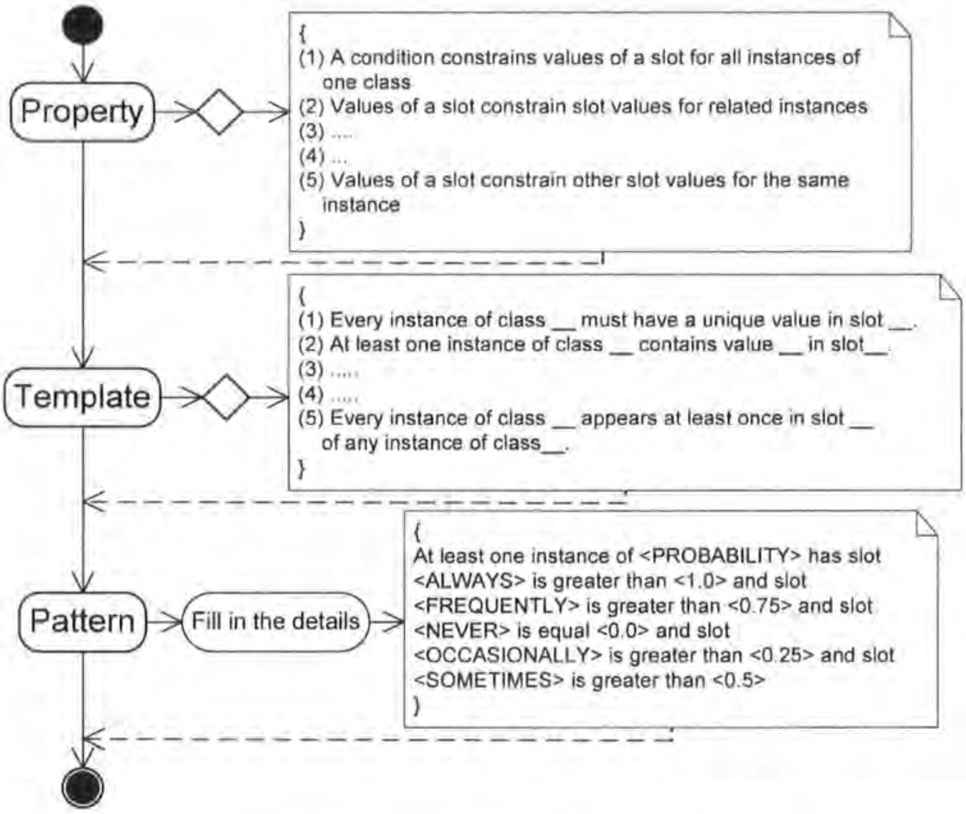


Figure 6.9: Method of Defining Constraints

For example, in the *Organisational_Knowledge* ontology, *Probability_Factor* has been declared as one of the *superclasses* used to define a constraint related to the instances captured within the ontology (see figures 6.2 and 6.3). In order to set the axiom constraint in the *Probability_Factor* class, there were three stages the developer had to complete as shown in Figure 6.6. First, the developer must select which description under *(property)* is suitable to describe the constraint. In this case, the description "*Values of a slot contain values for related instances*" has been used. Under *(template)* there were several modes to be selected to describe the values of a slot, in this case "*At least one instance of class (name) contains (value) in slot (name)*" was the appropriate mode to be used to declare the values within the multiple slots. *Pattern* is the final stage for the developer to fill in the actual values of the slots. The selection processes were based on experience, in order to declare the appropriate constraint, the developer may need to run through the processes a few times to obtain the optimum solution.

Figure 6.10 illustrates an example of the constraints declared in the Organisational Knowledge ontology as a result of using the EZPal tab. The constraints were built upon the syntax of the axioms statements, descriptions and range.

(a) The syntax describes the *Probability_Factor*.

(b) The probability values shown as direct instances in the *Probability_Factor* class.

(c) The direct instances were then selected to associate with a specific knowledge statement as depicted.

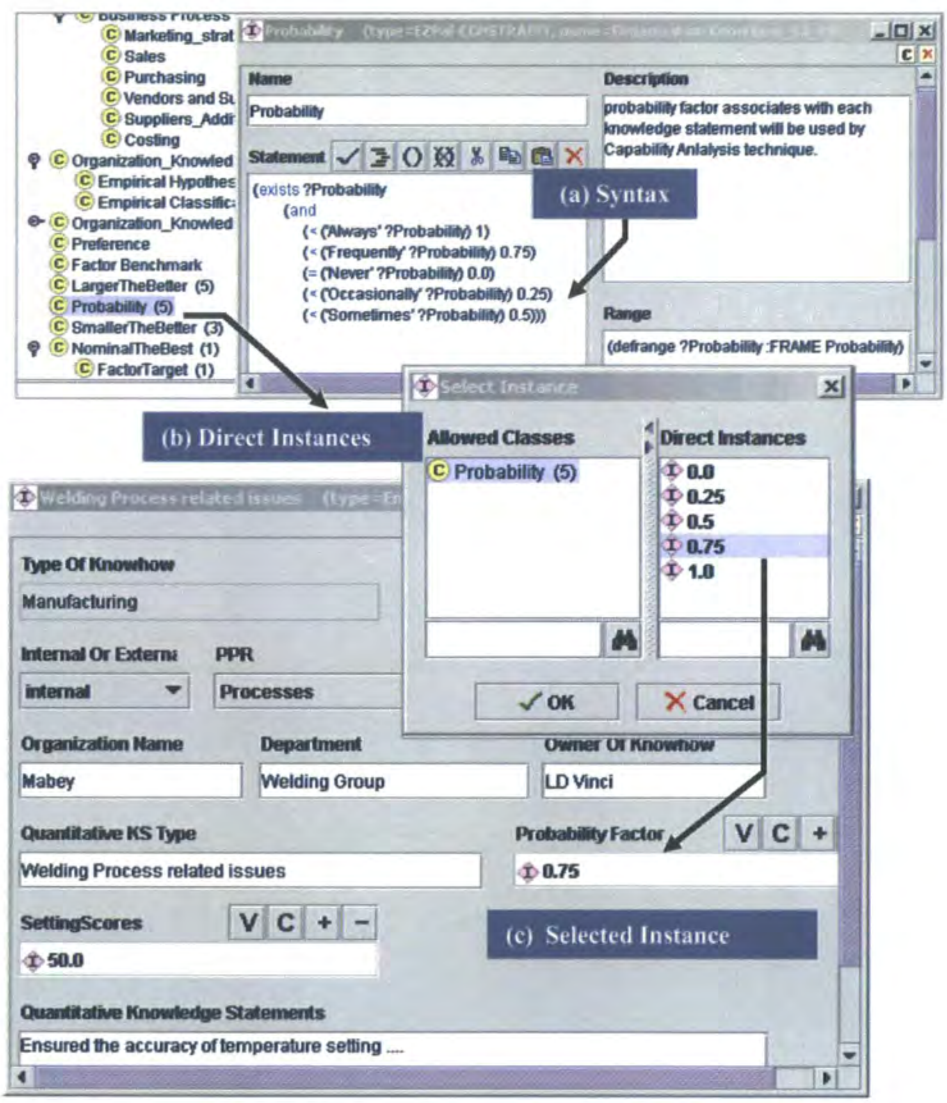


Figure 6.10: Example of Using Constraints

6.5 SUMMARY

In this chapter, an Organisational Knowledge Framework using Web-based technologies and information management systems for the collaboration of distributed teams in product development has been defined and developed. The development of the Organisational Knowledge Framework has been created as part of the tool for the product development integration architecture to support early design and manufacturing evaluations as described in Chapter 4. The novelty of this part of the work is the development of a manufacturing know-how data structure (Section 6.3.3) which has been constructed as part of an Organisational Knowledge Framework using an ontological approach towards capturing and reusing manufacturing knowledge (Section 6.3).

This chapter has also shown examples of ontology tools selection (Section 6.2.3). The selection of ontology knowledge editors is important too as there are more than ninety ontology editors available. Some of the editors dictate the use of a specific kind of knowledge domain. For instance, the design knowledge domain may need a particular type of editor feature which has the capabilities for geometric representation.

As a result of the theoretical and practical experiences in using ontology and knowledge management, this chapter has specified and created an Organisational Knowledge Framework compatible with aggregate process planning (Section 6.3). This chapter has also discussed the implementation of capturing and reusing manufacturing know-how using various ontological activities and the application of axioms and constraints (Section 6.4). With the development of user-friendly ontology editing software and automatic data exchange functions, the application of ontological approaches to exchange information across the WWW will be an essential aspect of the next generation of global knowledge management tools.

As for the following chapter, it presents detailed discussion and implementation of the product development integration architecture in linking the design and manufacturing domains through the applications of Web-based and enterprise integration technologies.

Chapter 7

LINKING DESIGN AND MANUFACTURING DOMAINS VIA WEB-BASED AND ENTERPRISE INTEGRATION TECHNOLOGIES

7.1 INTRODUCTION

As described in Chapter 1, the overall theme of this research is to create a collaborative environment and its associated components to bridge the gap of communication in the early product design and development stages via the application of a novel product development integration architecture as briefly highlighted in Chapter 4. Chapters 5 and 6 discussed the main development of the aggregate manufacturing models and an organisational knowledge-based system. This chapter will discuss the main integration architecture for linking design and manufacturing domains through the application of Web-based technologies and enterprise systems. This chapter focuses on the implementation issues of PDM, ERP and Web-based technologies. The integration architecture forms the theoretical backbone and defines the role of the system in supporting product development in a collaborative and knowledge distributed environment.

The integration architecture is designed to be used by product design, product development and manufacturing engineers to explore possible design alternatives in a Web-based environment. In practice, the integration environment can be used in a collaborative manner by vendors, original equipment manufacturers (OEMs) and suppliers with deployment of different ERP and PLM/PDM systems. One of the main features of the system is the data exchange mechanism which enables industrial standard data format (XML) to be reused with third party software systems.

In addition, this chapter also discusses and presents the implementation of a hybrid integration methodology which is a subset of the integration architecture to address the issues of utilising open source P2P, JXTA and PLM. The focus of the adoption of open source is to evaluate how a decentralised communication network can enhance the integrity of knowledge and data sharing for the collaboration of smaller companies and cooperation within the supply chain. The application of P2P methods was initiated at Cranfield. However, the author of this thesis has made significant contributions to several elements of the research and development issues such as:

1. security in P2P,
2. the use of distributed workflows, and
3. refining the connectivity of the infrastructure and testing.

A more comprehensive view of the development issues of P2P contributed by this author is shown in Section 7.4.2.

7.2 THE PRODUCT DEVELOPMENT INTEGRATION ARCHITECTURE

The proposed integration architecture is illustrated in Figure 7.1. The overall integration environment is categorized into *three layers*. The first layer is the enterprise system which consists of the PDM and ERP technologies and P2P for inter-enterprise collaboration. The second layer is the communication and data exchange mechanism, including JXTA, client/server and XML Parser. The third layer consists of the Manufacturing and Design Domains.

The architecture uses PLM systems to address design interoperability. This out-of-box solution provides the functionality of different designers at different locations to access the same design collaboratively. The architecture also supports STEP-based standards for geometric models. This standards-based collaboration can work in a global, distributed, and heterogeneous design environment. In addition, PLM offers *lifecycle* *management* and *versioning control* for the design and the ability to see the history or 'evolution' of a design through all its iterations. Thus, this allows geographically dispersed users to co-edit CAD geometry and related tasks dynamically.

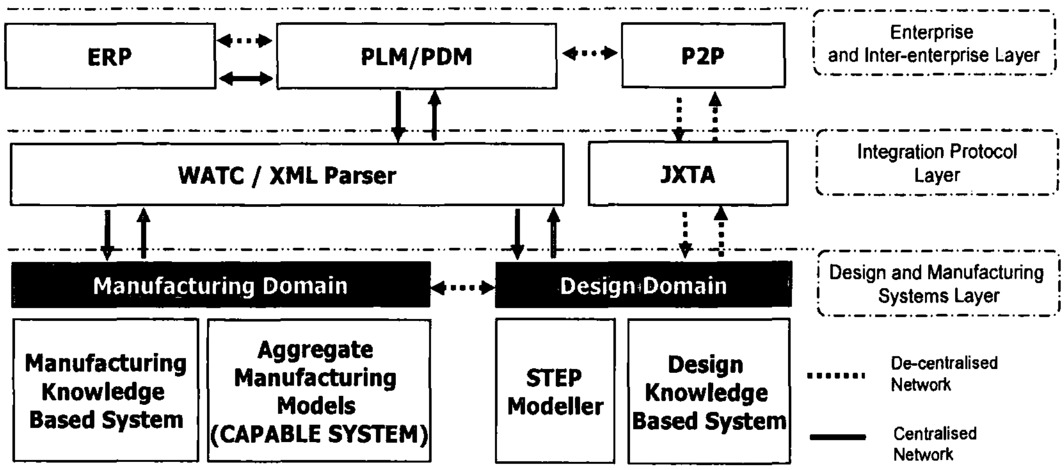


Figure 7.1: Overall System Integration Architecture in Product Development

7.2.1 The Enterprise and Inter-Enterprise Layer

The deployment of a PLM/PDM system provides an 'integration wrapper' for the entire integrated system. It supports an online distributed and collaborative environment with specific functions including *product data/document management, versioning control, workflows* and *lifecycle management.* The term integration wrapper denotes the ability to "wrap data and knowledge" from different domains into a common format, such as XML, so that a file can be shared within a distributed PDM environment and readily interpreted by using the terminology definitions of the ontology. The deployment of an open source ERP system is mainly used to generate capacity requirements planning based on the assembly and subassembly sequences of specific products. The P2P technology supports a de-centralised inter-enterprise collaboration which will be discussed in Section 7.4.

7.2.2 The Integration Protocol Layer

An XML Parser is deployed as the interfacing technology between a PDM system and the Manufacturing and Design Domains for data interchange. This enables interchanging portions of XML documents while retaining the ability to parse them correctly and, as far as practical is concerned, they can be formatted, edited, and processed in useful ways. The discussion of implementing the XML Parser is in Section 7.3.2. JXTA is an open source communication protocol for P2P network connectivity which is discussed in Section 7.4.

7.2.3 The Design and Manufacturing Systems Layer

The manufacturing domain consists of the CAPABLE system (Chapter 5) and manufacturing knowledge-based system (Chapter 6). The discussions of implementing the STEP Modeller and the Design Knowledge-Based System are based on the research work at Cranfield University. A CAD system Pro/Desktop is used as the solid modeller to display the image of the product through a PDM Visualization functionality. An Oracle database server is also deployed to handle requests, knowledge and model information as well as deploy PDM functionalities through the use of Java Database Connectivity (JDBC).

7.3 COORDINATION ISSUES OF THE ACTIVITIES WITHIN THE INTEGRATION ARCHITECTURE FOR THE TIME-DEPENDENT COMPONENTS

The integration of distributed and time dependent components requires a time synchronisation model. The time based co-ordination element requires the recognition of the time-dependencies of activities within a distributed team that use the stored data and knowledge. In general, a PDM system comprises; a "*Document Manager*" that contains a list of user defined *cabinets* to store data files, a "*Lifecycle*" function that defines the timing of the development stages and a "*Workflow*" function that determines what processes and interactions take place at each stage. Clearly, PDM functions can be used as a foundation for defining a time based integration wrapper as a time synchronisation model.

Figure 7.2 illustrates the coordination requirements for the design of a workflow, which can be differentiated into internal and external integration requirements (Becker et al. 2002). Lifecycle management is responsible for defining the development stages from conception through design, engineering, manufacturing, use and maintenance to disposal. The role of networking allows all users to interact with the Web services and ensures the collaboration of external and internal applications.

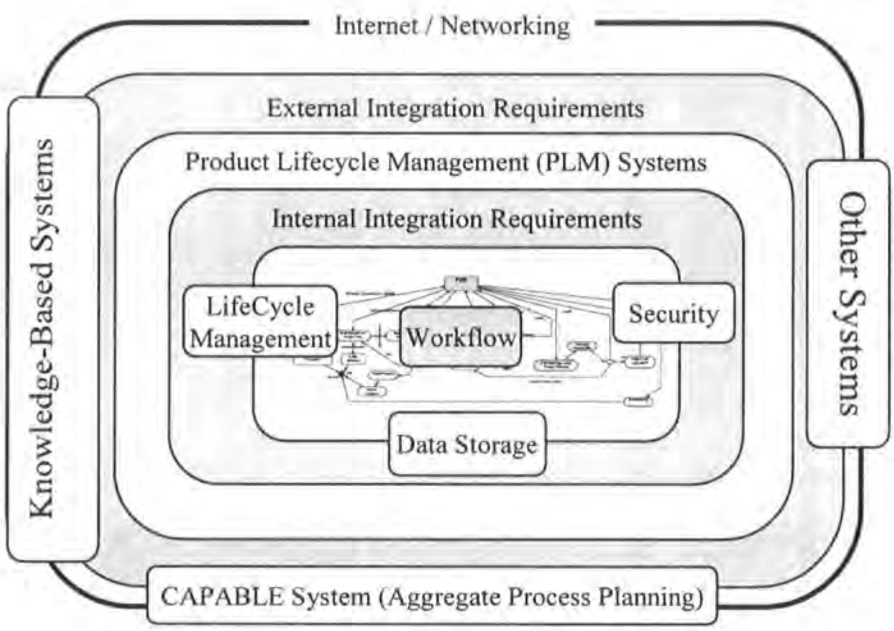


Figure 7.2: Coordination of Time Dependent Components

Internal integration requirements concern those systems and functions with which the workflow applications need to connect to. For example, the internal functionality of a PDM system is life-cycle management, data storage and security. External integration requirements exist with regard to systems that either invoke the workflow system from the outside (embedded usage) or systems that are invoked by the workflow applications, for instance, the Knowledge-based system and the CAPABLE System. Since work coordination involves external networks and external applications, another issue that must be addressed is security in the World-Wide-Web. One major advantage of using PDM systems for workflow management is that they automate the information flows between individuals carrying out business processes and it has two major implications for security.

1. Firstly, since the description of a workflow process explicitly states when each function is to be performed and by whom, security specifications may be derived from such descriptions and translated into static role-based specifications.

2. Secondly, since the Workflow functions are to be operated via Web-browsers, individual security rules can be enforced.

7.3.1 Workflow Activity Task Controller (WATC)

A novel "Workflow Activity Task Controller" (WATC) methodology has been defined to implement the *time based integration wrapper* concept in the interactions between generic types of PDM, ERP, Knowledge-Based systems and Process Planning functions. The methodology has been formalised in UML as shown in Figure 7.3. WATC sequences early design activities including *concept definition*, *design development*, *manufacturing knowledge sharing* and *automated aggregate plan* generation. WATC currently supports the following five early design stages:

1. Receive/Understand Customer Product Request and Formalise Design Specification.
2. Generation of Conceptual Design by the Product Development Team.
3. Distributed Review of the Conceptual Model and Addition of Manufacturing Knowledge and Constraints.
4. Deployment of Capability Analysis for the Prioritisation of Product Development Tasks.
5. Generation of Aggregate Process Plans (Routings) and Integrated Capacity Planning.

The core technologies behind WATC are methods to control the interactions of a PDM system, a Knowledge Based System and a Process Planning System. The implementation of WATC is centred on the lifecycle and workflow functionalities of the PDM. An example output of the iterations is shown in Appendix H.

The workflow starts with the customer's request for a new product or a change to an existing product as shown in Figure 7.3. All business processes are modelled graphically within the PDM system as flow charts, a sample screenshot of a workflow is shown in Figure 7.4, which indicates an email notification from a customer. The initial stage is adding customer historical information such as previous product specifications, customer buying experience and relationships. This can be done by invoking a knowledge-based system. The knowledge-based system consists of two separate modules, one is for the design knowledge management system that captures information related to product design and design standards, and the other is for manufacturing knowledge management to capture process and resource related knowledge.

1. The primary action of the workflow is to activate the process by assigning a task to make a connection with the KBS. All the information or relevant knowledge is stored or retrieved via a Windchill PDM Cabinet function (PTC Windchill 2002). The Windchill Cabinet function is used to store product centric information and provides a method of locating information within the PDM system.
2. The second stage of the workflow is to assign a concurrent task which involves notifying team members of the development team and issues requests to the appropriate personnel to enter conceptual design data.
3. The third stage of the workflow is to review the conceptual design.
4. The fourth stage is an XML Parser mechanism which supports the interaction of data reused of the CAPABLE and PDM systems.
5. The final stage involves capacity planning and implementation.

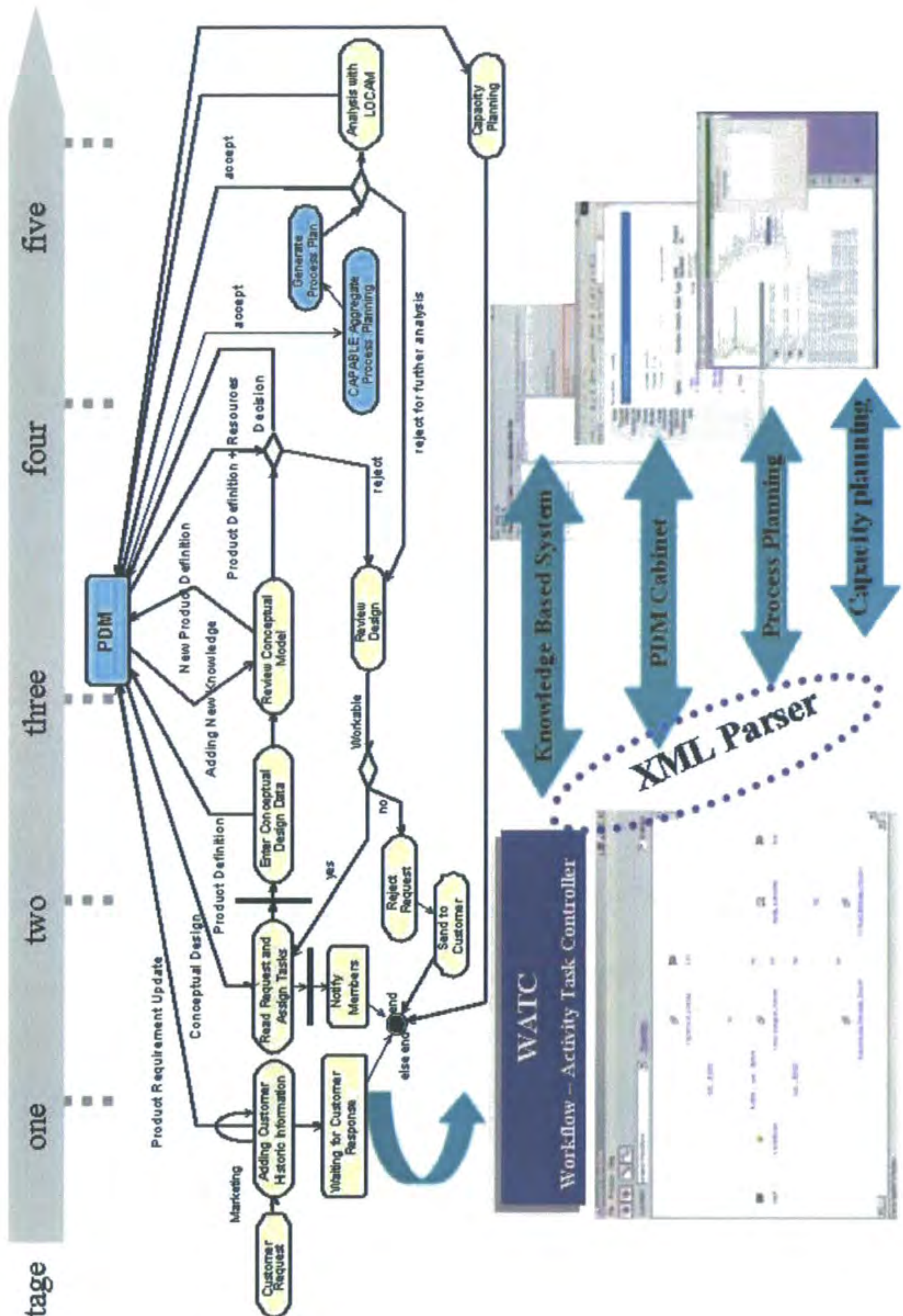


Figure 7.3: Time Dependency Scenario Using WATC Concept

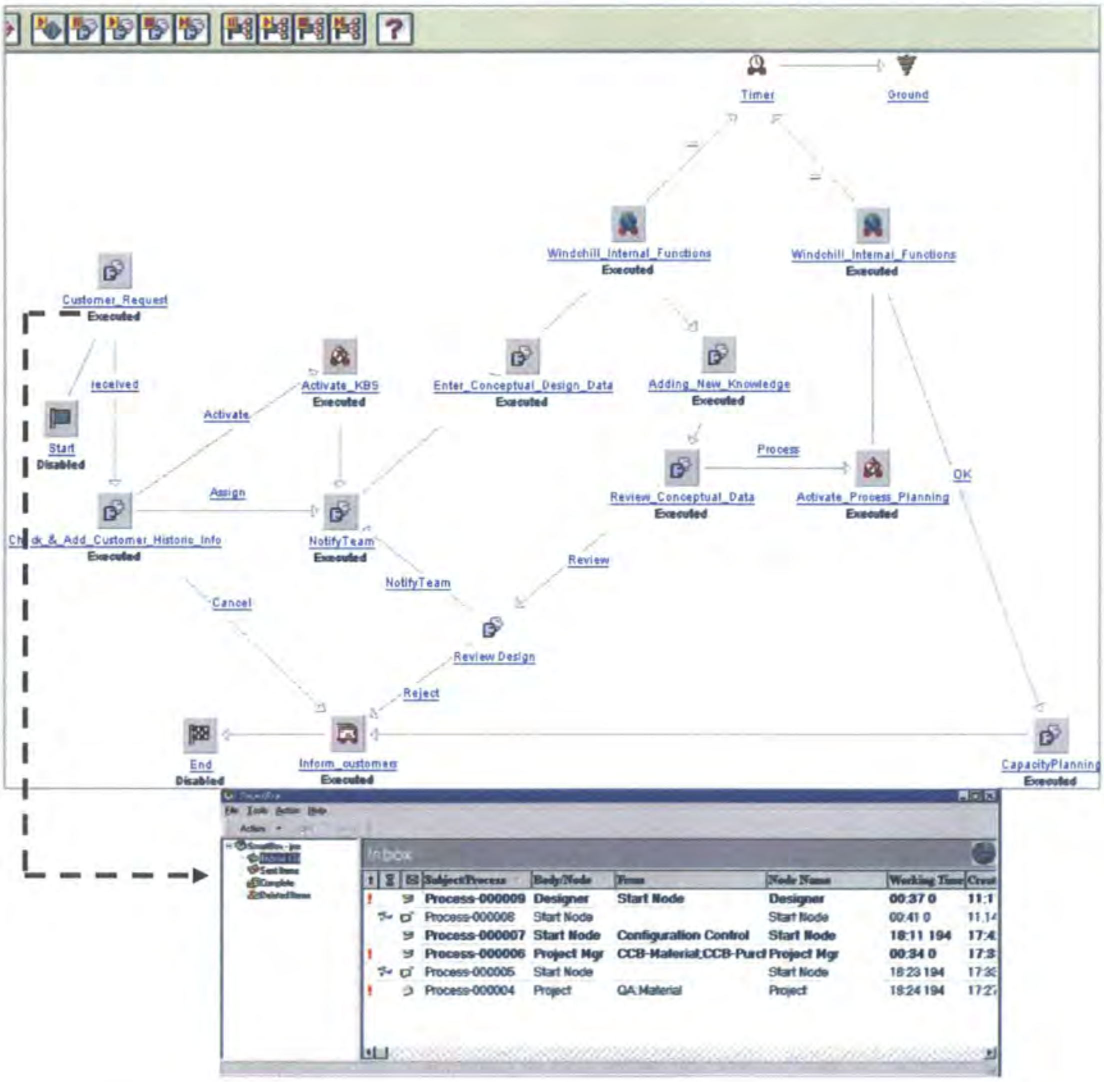


Figure 7.4: Sample Screenshot of the Windchill PDM Graphical Workflow and Email Notification Inbox.

7.3.2 The XML Parser Mechanism

The XML Parser is responsible for extracting manufacturing knowledge from the XML formatted knowledge file to be reused by the process planning engine in the CAPABLE system. With the attachment of updated historical information and manufacturing knowledge, a new product definition will be generated. The product definition will be delivered to the CAPABLE system to obtain preliminary process plans. The purpose of the CAPABLE system is to allow alternative process plans (or routings) for custom parts to be generated, evaluated and improved based upon estimated *manufacturability* before committing to a fully specified product model and supplier. The new process plans (routings) are then delivered to the PDM system for Plan/Review.

7.3.2.1 Methods of Creating the XML Parser

The advantage of using an XML-formatted file is that, there is a whole range of generic XML tools available to create an XML Parser for extracting the information and translating it into the required format of a proprietary tool to re-use. In this case, an XML Parser has been created for transferring the stored knowledge to the CAPABLE System. An XML Parser has been created based on the Java programming language. Figure 7.5 illustrates a UML Activity Diagram (Schmuller 1999) to represent the algorithm of a Java-based XML Parsers' internal methods that are used to read and extract the information (of data type string) and translate it into the format of a third party software system. The illustration represents two specific roles, the initial role is to *Prepare XML Metadata* and the second role is to *Extract XML Metadata*. Transition can take place from one role to another. One further aspect learned from creating the XML Parser is that the developer must understand the interface provided by the XML Parser (Stevens 2003).

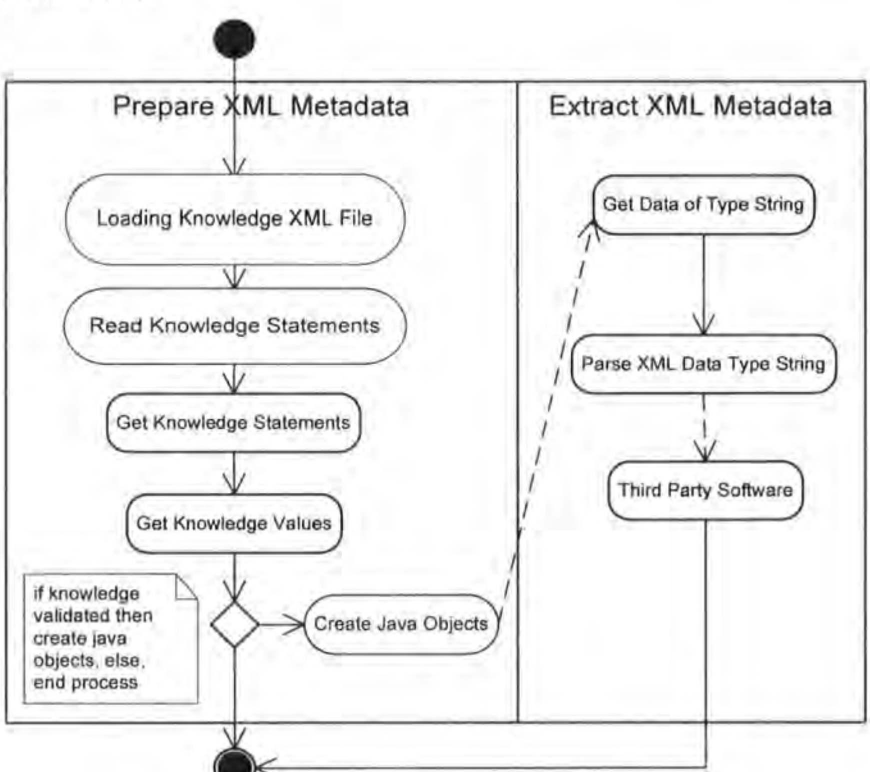


Figure 7.5: Algorithm of a Java-based XML Parsers' Internal Methods

- Prepare XML Metadata

The first stage to *prepare XML metadata* is to *load* the XML based knowledge file as shown in Appendix I (I1). The KBS captures knowledge statements related to product design, process capability and resources requirements. The main objective of the XML Parser is to translate all these types of information to a specific data model into the CAPABLE System.

The next stage of the functional requirements for the XML Parser is to *read* the XML-formatted statements using the *walkTree ()* function as shown in Appendix I (I2). Appendix I (I3) illustrates an example of the java statements of reading XML based knowledge statements into the factory resource model using *read (resource)* function. The Parser will *get* the statements and store them in a temporary *stack* as shown in Appendix I (I4), which indicates an example of '*Subject type*'. Once this chunk of information is completed, the Parser will then initiate the *get knowledge and get value* as illustrated in Appendix I (I4).

- Extract XML Metadata

The next step is to create Java objects. This is the stage where XML statements are translated into *java objects* as shown in Appendix I (I5). The way to identify the correct statement being extracted into the right place is by defining the *tag name* inside the XML Parser which must match the *Markup* defined within the XML documents. The previous illustration in Appendix I (I4) also shows an example of creating the java object 'Subject Type'. The name '*Subject Type*' corresponds to the name tag '*Subject Type*' as defined within the XML document. XML documents are composed of *entities*, which are storage units containing text and/or binary data. Text is composed of character streams that form both the document character data and the document markup. *Markup* describes the document's storage layout and logical structure. A simple example of the work described here for representing manufacturing knowledge statements in XML is shown in Figure 7.6.

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE document (View Source for full doctype...)>
  - <document>
  - <Knowhow>
    <Date_Prepared date="28/02/2004" />
    <SubjectType subject="RoboticWelding" />
    <FPR>Processes</FPR>
    <Probability Factor>0.75</Probability_Factor>
    <Knowledge_Type knowledgetype="BestPractice" />
    <Organization_Name organization="Mabey & Johnson" />
    <Owner_of_Knowhow ownerknowhow="Andy Chriswell" />
    <Qualitative_KS_Type>Factory Resources</Qualitative_KS_Type>
    <Qualitative_Knowledge_Statements qualitativeKS=" Do not weld on
  coated metals, such as galvanized, lead, or cadmium plated steel, unless
  the coating is removed from the weld area, the area is well ventilated,
  and while wearing an air-supplied respirator. The coatings and any
  metals containing these elements can give off toxic fumes if welded." />
  - <Knowhow>
```


Figure 7.6: Algorithm of a Java-based XML Parsers' Internal Methods

7.4 A PEER-TO-PEER FRAMEWORK FOR ENTERPRISE COLLABORATION IN PRODUCT DEVELOPMENT

7.4.1 Background of Peer-to-Peer

P2P applications address the needs of de-centralised organisations to collaborate and share knowledge and files regardless of geographical location (Bond, 2001). The principle of P2P has been around for a long time, and is today implemented in a number of applications such as instant messaging and file-sharing as described in Section 2.10. The advantages offered by P2P applications are (Bond 2001; Penserin *et al*. 2003):

- No single point of failure, the network is alive as long as one peer is on-line
- Distributed sharing of bandwidth, storage and processing power, so the system becomes more powerful as more users attach

- Lower running cost due to the lack of servers or high bandwidth central nodes, as well as
- Maintaining individual control of the shared knowledge.

P2P groups can be used to create profiles of the peer, and also more importantly of the peer's list of contacts within different domains. These profiles can be used within the network to search for and assess people's competences, interests, and memberships of trusted groups, and can aid in the construction of new relationships based on commonalities and third party assessments. However, a pure P2P architecture can reduce the performance of the system (Penserin *et al.* 2003). As such this research has developed an idea of using a hybrid P2P network as describe in the following.

In order to support the advantages of client/server systems with the independence and interoperability of P2P systems, a hybrid system where "super peers" act as peers to the extended P2P network and as a server to the enterprise's internal peer network is used. In addition, rendezvous peers can be assigned to manage some of the peer information assigned to particular peer nets or projects (Penserini *et al.* 2003; Leuf 2003). These hybrid systems have the potential for high-performance de-centralised services.

In terms of searching information and query, the hybrid P2P network uses a distributed index method (Clark *et al.* (2000), (2002)). According to Risson and Moors (2006), in order to find a file using the distributed method, the requesting peer first checks its local table for the node with keys close to the target. When that node receives the query it too checks for either a match or another node with keys close to the target. Eventually, the query either finds the target or exceeds time limits. The query response traverses the successful query path in reverse, depositing a new routing table entry (the requested key and the data holder) at each peer. The insert message similarly steps towards the target node, updating routing table entries as it goes, and finally stores the file there.

7.4.2 Overall Peer-to-Peer Framework for Enterprise Collaboration

In today's economy, effective information and data sharing is directly linked to the competitiveness of an enterprise. Whether it is a single enterprise or a network of organisations, effective collaboration and data distribution is an important requirement. This is particularly true for smaller companies such as Small and Medium-Sized Enterprises (SMEs), who nevertheless, find it difficult to meet the cost of software, the need of high power computer systems and its maintenance. The costs are a barrier for smaller companies who want to work with larger companies who may require large scale software systems. One example is the Product Lifecycle Management system (CIMdata 2002). PLM is an expensive software system and is mostly operated by large organisations. Apart from the cost of the software system, there is an additional disadvantage for SMEs to collaborate with the OEMs (Original Equipment Manufacturers) or global enterprises; SMEs have to store all relevant product development data within the PLM system deployed and managed by the OEMs. The SMEs in this case do not have any degree of autonomy in such a restricted environment.

One of the main applications of PLM systems is the control of product development data within a centralized network throughout its product lifecycle using their native *workflow* function. Most of mainstream PLM systems operate in an inherent client/server environment which is overly complex to set up and requires a long period of time for customisation. The system's main strengths are in the deployment of the workflow tools. However, it has been identified that individual workflow elements and any customisation carried out within a PLM system cannot be directly reused by another PLM system (Aziz *et al.* 2005). On the other hand, the *RosettaNet* standard (Yendluri 2000), implemented in PLM systems, can decrease the amount of customisation needed, but not eliminate it, as *RosettaNet* only provides a subset of the messaging and data models of the inter-enterprise link.

The aim of this section is to discuss a hybrid P2P framework for enterprise collaboration as shown in Figure 7.7 which is a subset of the product development integration architecture as described in Section 7.1. The main technologies used in the framework are the P2P decentralized network (Penserin *et al.* 2003) and open source workflows implementing *RosettaNet* and *JXTA* (JuxtaPose) (Gong 2002) for network connectivity. The principle and advantages of P2P have been described in Section 2.10. One further aspect of P2P systems is to distribute the main costs of sharing data, computing power, network bandwidth and storage capacity (Bond 2001). This is particularly suitable for SMEs that do not need powerful, expensive servers and provides operating autonomy in terms of data sharing with the PLM system used by the OEMs.

The schematic representation of the overall hybrid P2P framework in Figure 7.7 is explained as follow:

(i) The integration architecture as discussed in Section 7.1. (Durham)

(ii) To set up an inter-enterprise collaboration using P2P, this will be discussed in Section 7.4.3. (Durham and Cranfield)

(iii) Initiate a collaboration project using workflows as discussed in Section 7.4.4. (Durham)

(iv) The P2P collaboration supports by 2-tier architecture (as discussed in Section 7.4.5) with JXTA for network connectivity and JDBC for databases. (Cranfield)

(v) The design and manufacturing knowledge based-systems are defined using different standards such as STEP (Cranfield for solid modeller) and open source OWL (an advanced version of RDF), the knowledge-based systems were developed based on RDF.

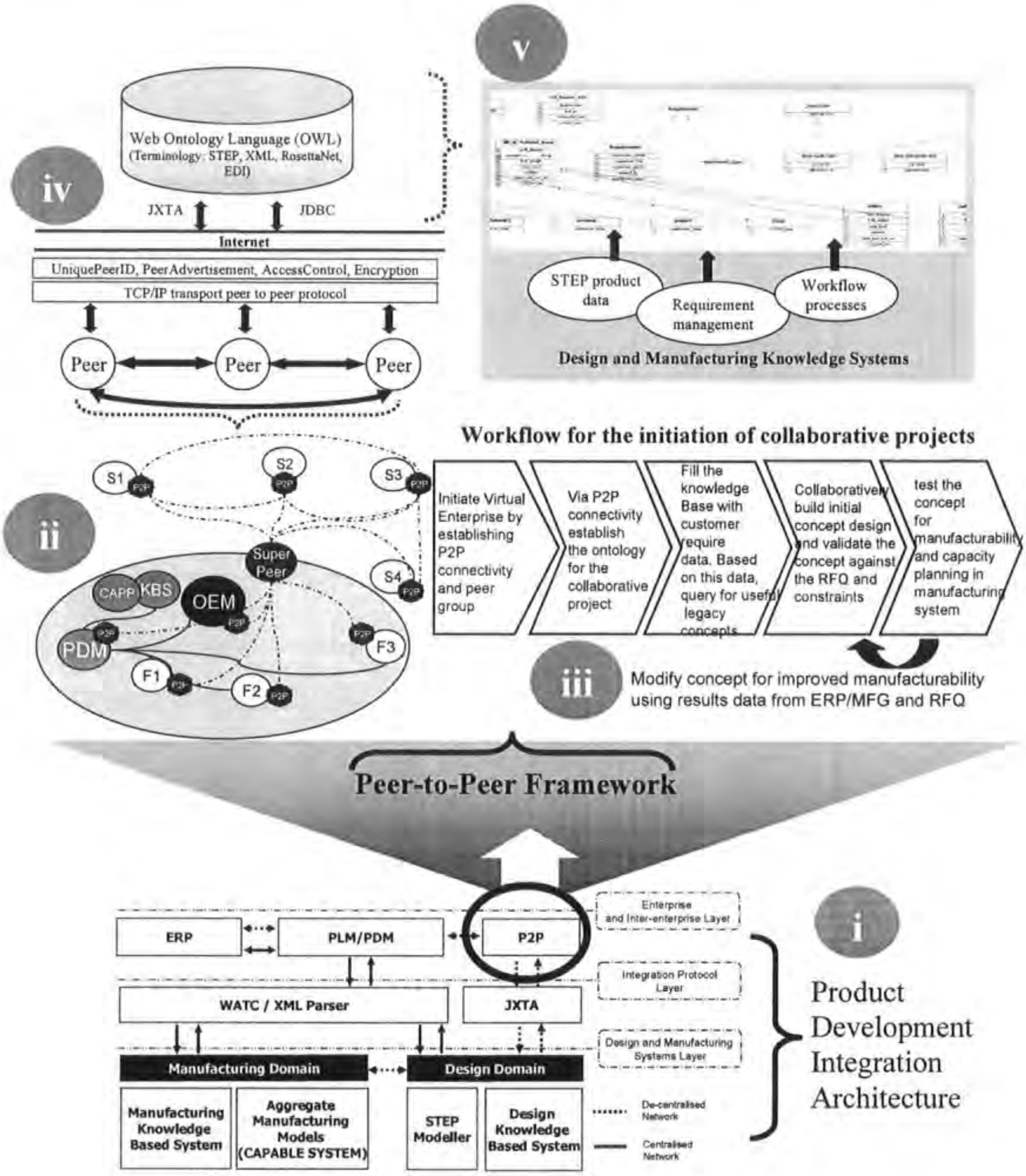


Figure 7.7: Overall Hybrid P2P Framework for Enterprise Collaboration

7.4.3 Inter-Enterprise Collaboration Framework Using P2P

Figure 7.8 illustrates the configuration and characterisation of the product development collaboration processes between SMEs and OEMs or other large organisations using P2P. S1, S2 and S3 are referred to as SMEs. In the OEM schematic representation, F1, F2 and F3 are the internal functions or divisions of the organisation, for example, product design, process planning and marketing. A Process Planning system and a

Knowledge-based System (KBS) are used as examples of different types of software deployed within the OEM via the PLM system.

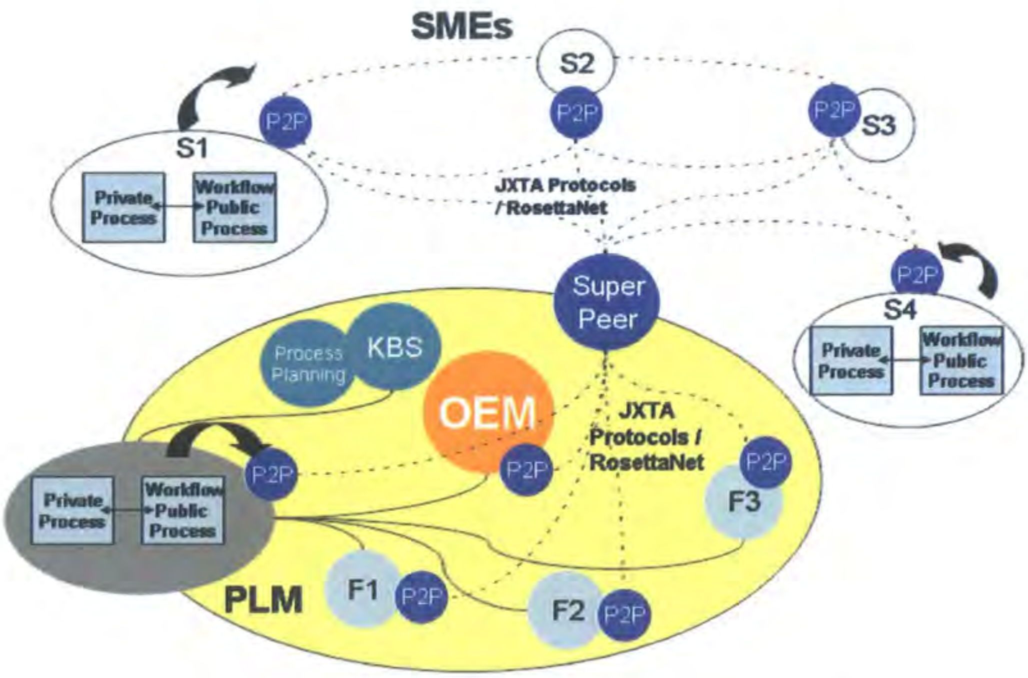


Figure 7.8: Inter-Enterprise Collaboration Using P2P

The framework indicates the basic infrastructure of *hybrid* P2P networking (Fiorano Software 2003), employing both the JXTA communication protocol and *RosettaNet*. In a hybrid P2P system, the control information is exchanged through a *central server* (the SuperPeer node). The control server acts as a monitoring agent for all the other peers and ensures information coherence. The basic infrastructure builds on the already available open source P2P solutions. A workflow technique is being deployed to ensure better control of data and file sharing in order to distribute information to the right people, at the right time. Information requirements are set out during tasks delegation throughout the product development processes. The PLM system used by the OEM is located on a local client/server centralised network.

The "P2P" nodes illustrated in the Figure 7.8 act as information connectivity nodes. Connectivity is achieved using an open source implementation of JXTA. This element was made partly for the security aspect, but also because JXTA implements a unique but anonymous identification mechanism for peers and rendezvous peers. Also "advertisement" is implemented for all peers that give information of one peer to other peers. Rendezvous peers can act as managers for peer groups and store the peer advertisement for the group for distribution to other P2P networks. Implementation of JXTA is in standard Java 2. The user functions of JXTA include the following:

- user interface queries
- project management of collaborative groups
- group chat and instant messaging

Additionally, the system's settings enable enterprises and users without static addresses to collaborate using dynamic addressing, as well as the users' ability to work offline which cannot be done with Web-based systems. Figure 7.9 shows an alternative configuration of the framework using SuperPeer nodes as communication connectivity. S1 consists of two subdivisions represents by S1a and S1b which forms its own *peergroup*. Similarly, both S2 and S3 have the same configurations. This shows the following advantages:

- Flexibility of forming a larger virtual network as members increase.
- Security is enhanced as the 'super-peer' node acts like a centralised control server and this allows users or peers to join a specific *peergroup* after a set of credentials is verified.

However, it introduces scalability issues, as peer members increase so does the space needed in the *Superpeer* to hold the information about all the users. Consequently, as members increase so do requests of information and thus, it becomes bottlenecked and its performance will be reduced.

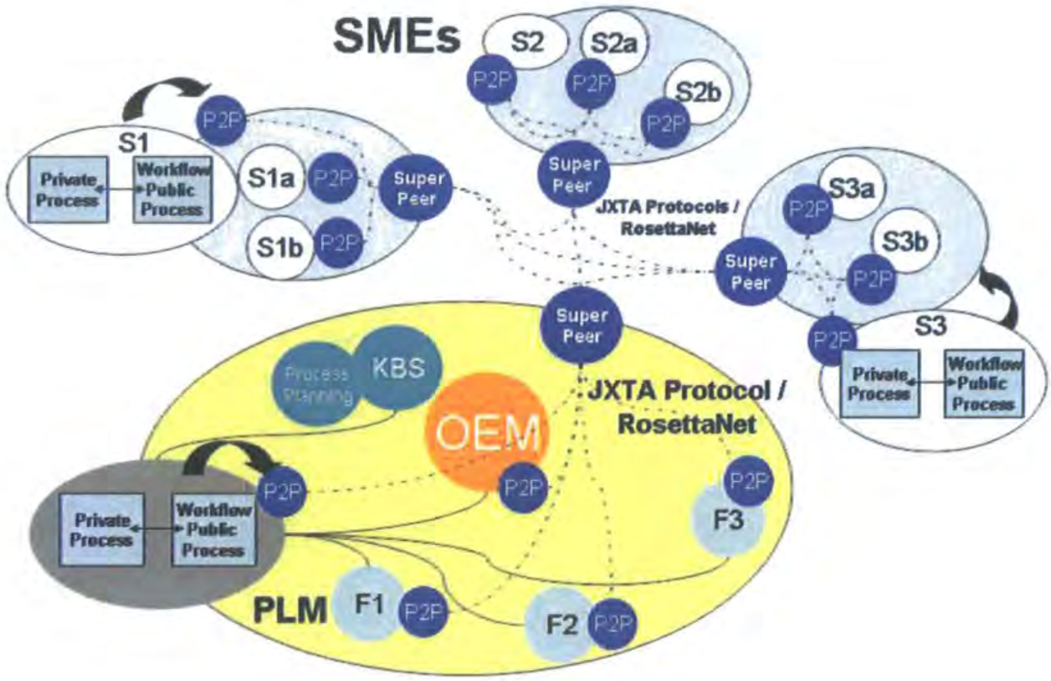


Figure 7.9: Alternative Configuration of Inter-Enterprise Collaboration Using P2P

7.4.4 Distributed Workflow

Frequently, process workflows are distributed collections of activities that involve groups of individuals at disparate locations. To coordinate these tasks, a process support system should provide for distributed process execution and integration with various tools across networks. Traditional workflow adopts a centralised client/server approach to enact processes such as the native workflow function that is usually provided by the PLM system of the OEM. For a decentralised network, this research uses workflows to implement RosettaNet Partner Interface Processes (PIPs) (Yendluri 2000). Figure 7.10 depicts a novel solution contributed by Durham University for a distributed workflow that is applicable to SMEs and OEMs collaboration. The use of WebLogic Integration (Yendluri 2000) implements standard RosettaNet PIPs through "public workflows" (also known as collaborative workflows). A "public workflow" provides the interface to other collaborative partners, while "private workflows" are used to interface to back-end systems in order to generate and respond to messages. Figure 7.10 shows the process by which

PIP workflows pass messages between collaborative partners. Generally speaking, RosettaNet-oriented workflows process messages as follows:

1. SME1's private workflow initiates a RosettaNet message. Data is retrieved and formatted into a RosettaNet message structure, the appropriate PIP is determined, and the message is forwarded to the public workflow that implements the SME1's workflow in the PIP.
2. The public workflow process creates the appropriate RosettaNet message. The message is sent to the public workflow of the OEM.
3. The OEM's public workflow receives the message, processes the header information, and then passes validated customer information and message content to the appropriate private workflow process.
4. The OEM's private workflow process resolves message content and generates a reply. The reply is processed into a RosettaNet message structure and passed back to the OEMs public workflow process.
5. The OEM's public workflow process creates a RosettaNet reply message and sends it to SME1's public workflow process.
6. The SME1's customer public workflow process receives the reply message, processes header information, and then passes validated OMEs information and message content to the appropriate private workflow process.
7. The private workflow process resolves content of the reply message.

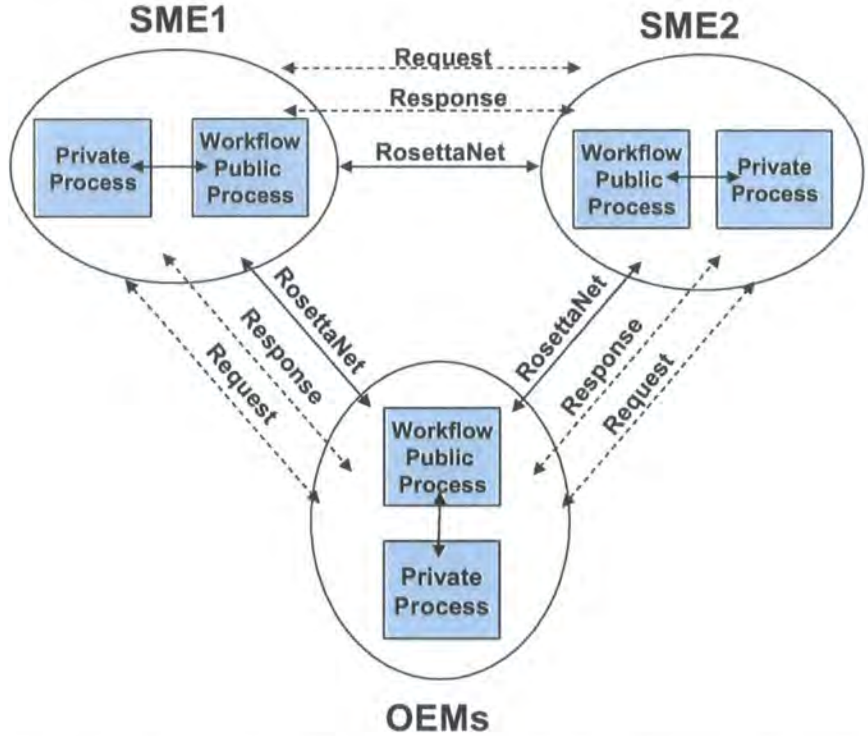


Figure 7.10: Distributed Workflow Diagram for SMEs and OEMs Collaboration

7.4.5 The 2-tier Communication Architecture in P2P Networking

There are several P2P communications protocols available, the most common ones are:

- JXTA
- Napster
- Gnutella
- AIM

The applications of these protocols for peer-to-peer systems are usually designed to deliver a single type of network service, for example, Napster for music file sharing, Gnutella for generic file sharing and AIM for instant messaging. For implementation purposes, JXTA was adopted as the protocol for the P2P framework because it is XML-based and is open source. It was initiated and developed by Sun Microsystems, Inc, and the company intend to standardise this technology for peer-to-peer networking. JXTA (Gong 2002) is a set of networking protocols similar to HyperText Transfer Protocol (HTTP) (http://www.w3.org) and TCP/IP (http://www.protocols.com). The JXTA layer sits between the networking stack and the application stack, handling peer-to-peer communications. The JXTA platform standardizes the manner in which peers:

- Discover each other
- Advertise network resources
- Communicate with each other
- Cooperate with each other to form secure peer groups

The framework described in this thesis uses the two-tier solution shown in Figure 7.11, which enables enterprises to establish connections between them in real-time. The first layer uses the JXTA communication protocol for XML-based messaging as well as the "P2P" nodes connectivity. The second layer uses RosettaNet mainly for distributed workflow and XML-based Electronic Data Interchange (EDI).

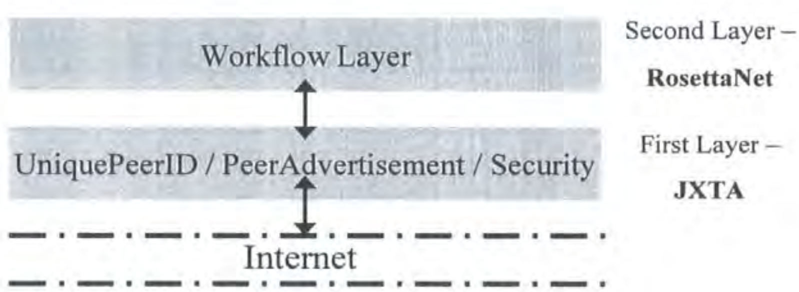


Figure 7.11: The 2-tier Communication

7.4.5.1 Security Issues in P2P Networking

Security is one of the main challenges in the implementation of P2P, due to the lack of a centralized control server. The security aspect represents a novel contribution to the P2P architecture by Durham. Current P2P applications are designed mainly for home users to file share for trading music and movies. These applications are not written with information security in mind and offer no encryption for sensitive information.

To address the security issues, Sun Microsystems created an infrastructure called Project JXTA (http://www.jxta.org), which allows programmers to use a common library when creating new P2P applications, by providing a robust, secure, interoperable Applications Programming Interface (API). One of the security features in JXTA is the implementation of *AuthenticationCredential* which allows users or peers to join a *peergroup* after a set of credentials has been verified.

Authenicationcredentials are used by JXTA Membership Services as the basis for applications for *peergroup* membership. The *AuthenticationCredential* provides two important pieces of information:

- the authentication method being requested
- Identity information which will be provided to that authentication method.

Furthermore, JXTA also allows developers to implement *PasswdMembershipService* which allows a Membership Service to be created based on a password scheme.

7.5 SUMMARY

This chapter has presented the product development integration architecture which has outlined the links between design and manufacturing domains by utilizing advanced Web-based and enterprise integration technologies (Section 7.2). A novel "Workflow Activity Task Controller" (WATC) methodology has been defined to implement the *time based integration wrapper* concept in the interactions between generic types of PDM, ERP, Knowledge Based and Process Planning systems and has been formalised in a UML activity diagram (Section 7.3). The significance of using XML Parser to support systems interoperability and data exchange has also been highlighted (Section 7.3.2).

Information is often regarded as the most important asset for a company. Technologically and financially efficient methods to transfer and store information are the future strategy for a company to compete globally. With the current issues of P2P security to be remedied, the future of business applications in P2P looks promising. Rather than spend excessive money on expensive hardware and software, businesses could utilise the P2P technology to build a distributed network of workstations. The workstations would each maintain a small chunk of data locally. As a result of this investigation and research, a hybrid integration methodology which is a subset of the product development integration architecture using P2P networking as the core technology has been defined (Section 7.4). The application of P2P methods was initiated at Cranfield. However, the author of this thesis has made the following contributions:

1. Refining the connectivity of the P2P infrastructure and testing (Section 7.4.3)
2. The use of distributed workflows (Section 7.4.4)

3. Security aspect in P2P (Section 7.4.5.1)

The aim of using P2P is to demonstrate the use of decentralised network for product development collaboration between smaller companies and larger corporations. The framework also shows its flexibility that can be adapted into different network configurations.

In Chapter 8, two case studies will be presented and discussed. The testing of the case studies are based on two individual products as introduced in Chapter 4. The first case study replicates a centralised client/server environment supporting the design of steel panel bridges for rapid assembly on-site. The second case study is focused on creating a virtual enterprise collaboration to compare the applications of a centralised PDM and decentralised open source solutions.

Chapter 8

TESTING AND RESULTS

8.1 INTRODUCTION

This chapter presents the results of the two industrial case studies, in particular, the various software components used in the product development integration architecture and how they can be used in a distributed and collaborative environment to support early product development processes. The prototype software systems are the in-house developed process planning system and knowledge-based system together with major enterprise systems, the PLM and ERP. The two independent case studies are designed to replicate the integration environment and to prove:

- The fundamental hypothesis of bridging the gap of communication in early design and manufacturing activities.
- The logic behind the methods of demonstrating the practicality and its potential benefits for industrial applications.
- The flexibility of the methodologies which can be used in different product designs and manufacturing environments.

Due to the commercially sensitive nature of the information, all examples use synthetic data based upon real examples of the participating industrial collaborators.

8.2 IMPLEMENTATION

The application of Protégé2000 was to create the Organisational Know-how KBS system to capture real data in design and manufacturing activities. The KBS developed within Protégé2000 is different from the conventional software system in terms of user interactions and interfaces. It does not contain an *execute file* to invoke the KBS. Instead, the user needs to adapt and familiarise the Protégé2000 environment. The knowledge-based system can only be executed within the software tool itself.

The in-house aggregate process planning system, CAPABLE, has been further developed to capture data related to the collaborators' product design, process capabilities and factory resources. An XML Parser data exchange mechanism has been implemented to convert XML-based knowledge statements to support process planning in product design decision making. Having created the XML-based knowledge file, the next stage is to invoke the PDM system and use the 'create document' function to load the knowledge XML-based file into the PDM document storage cabinet. This allows the document to be re-used by other users based upon request. The way to make use of the document is simply by invoking the PDM 'check-out' function, which loads the document into the user's local web-server.

To evaluate manufacturability the implementation uses an open source tool, the Compiere ERP (Compiere CMPCS for manufacturing). Compiere CMPCS covers all manufacturing activity within the various types of production environments. The main modules are Resource Planning, Production Plan & Demand Management, MPS, MRP, CRP and Manufacturing Orders & Repetitive. In order to use the software more effectively the user must customize the system to suit the needs of the work that is being undertaken. One further aspect of the software is that apart from the *import* function which can be used to download the process plan, the majority of the information such as weight and height of the product feature was manually entered.

During the course of the research, it has been established that the use of a centralised PDM system would be more beneficial if it is diverted to an alternative approach in sharing information with smaller companies. As a result of further research, a P2P networking approach seems to be the future strategy for information sharing and collaboration between larger corporations and smaller companies.

8.3 TESTING AIMS AND REQUIREMENTS

The testing is to accurately monitor the various components built to bridge the discontinuity of communication within product development teams using Web-based technologies and internet-based management systems. Further perspectives of the results demonstrate the applications of new trend technologies related to manufacturing knowledge management, data capture methodology in an open source environment and the use of an industrial standard data exchange format.

The product development integration architecture has been developed to demonstrate the application of PLM in managing manufacturing data in a collaborative and distributed environment. An in-house process planning system, CAPABLE, has been further developed to capture data related to the collaborators' product design, process capability and factory resources. In addition, an XML Parser data exchange mechanism has been implemented to convert the knowledge statements captured in the KBS to support process planning.

The testing criteria, however, are limited to selected processes and product features. The selected components are sufficient to create the demonstration systems. For instance, the testing example of Mabey & Johnson's Bailey bridge is only a single panel. The ArvinMeritor vehicle-door-latch assembly is limited to a few of its components and selected assembly processes. Another critical criterion is the deployment of the PLM and ERP systems, both of the systems can be executed remotely through an external network. For simulation, the testing was carried out between Durham University and Cranfield University rather than using the industrial collaborators' networks. Thus, in this case, the PLM and the ERP systems can be stored at Durham and Cranfield. The testing and analysis can be used to evaluate the overall system in real-time.

The tests were carried out in a single user access environment. However, the architecture also supports multi-user access using PLM and P2P technologies. PLM allows simultaneous reading and writing of the same data in the repository which supports concurrent engineering practices. This is done by locking parts or the entire database while the data are being updated. Furthermore PLM supports access authorizations and hence data inconsistencies can be avoided and relationships between data maintained. As in the P2P application, connectivity is supported by individual super-peer node and hence multi-user is one of the main strengths of the architecture.

8.4 CASE STUDY 1: THE MABEY & JOHNSON STEEL BRIDGE PANEL

8.4.1 Objectives and Aims

The objective of the test was to demonstrate the various software components, and the original concept of the proposal to bridge the gap of communicating early product ideas to support product development processes. The aim of the evaluation was to test:

1. The technical feasibility based upon the data supplied and collected from the industrial collaborator, Mabey & Johnson Ltd., and
2. The WATC method to coordinate early design activities using PDM and ERP technologies, the organisational knowledge-based system and the aggregate process planning system.

The exercise concerned the evaluation of a Bailey's single steel bridge panel at the conceptual design stage and how decision support can be enhanced.

8.4.2 Application of the CAPABLE System

8.4.2.1 The Product Description

Figure 8.1 shows an early design configuration of a single steel bridge panel. The bridge panel was modelled using U-shaped steel beams and solid rectangular steel plates. The bridge panel was constructed from 14 steel beams (4 horizontal, 8 diagonal and 2 vertical), 6 steel plates and 8 square slots. The overall dimension of the bridge is approximately 3m by 2m with an approximate weight of 130kg. Assembling the bridge panel is a complex task. The number of welding processes required to assemble the panel requires more than 240 procedural steps. An example of the type of weld is shown in Figure 8.1. Table 8.1 summarises the description, material types, requirements and the weight of individual items used in constructing a single steel bridge panel. Figure 8.2 illustrates the model of a bridge panel with EBoM (Engineering Bill of Material) configurations and the example product model modelled within the CAPABLE system. The product model represents a conceptual design stage of a single Bailey's steel bridge panel.

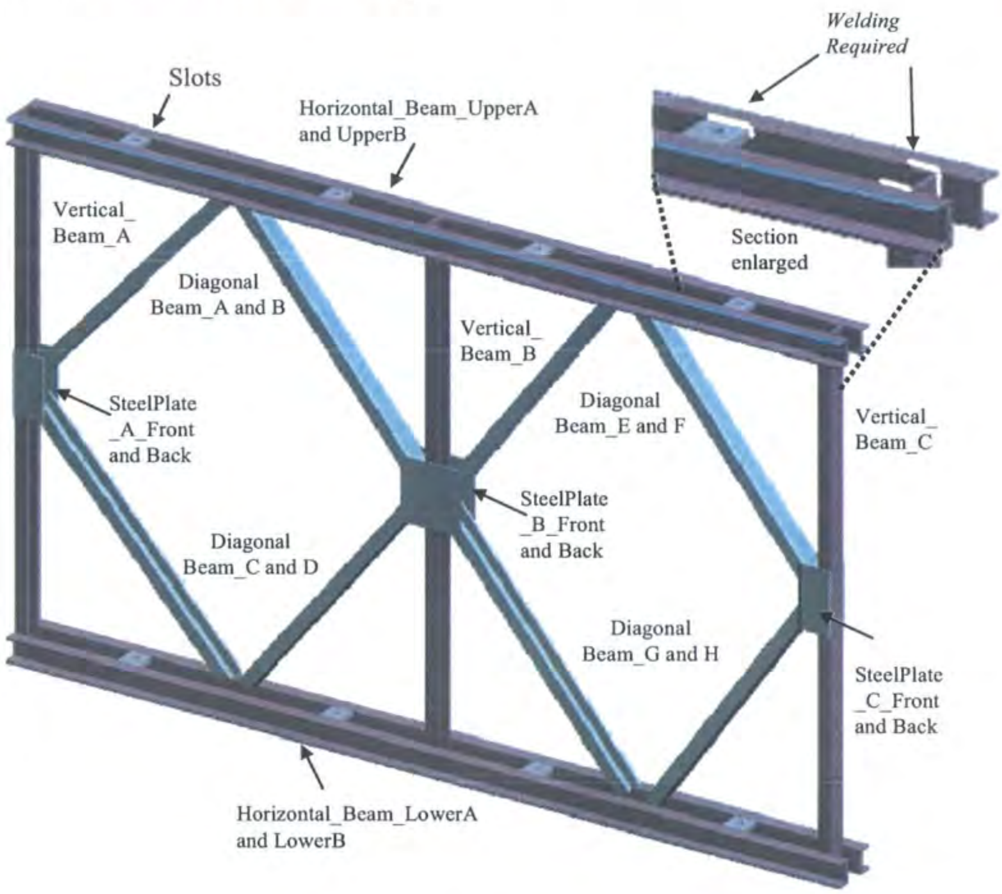
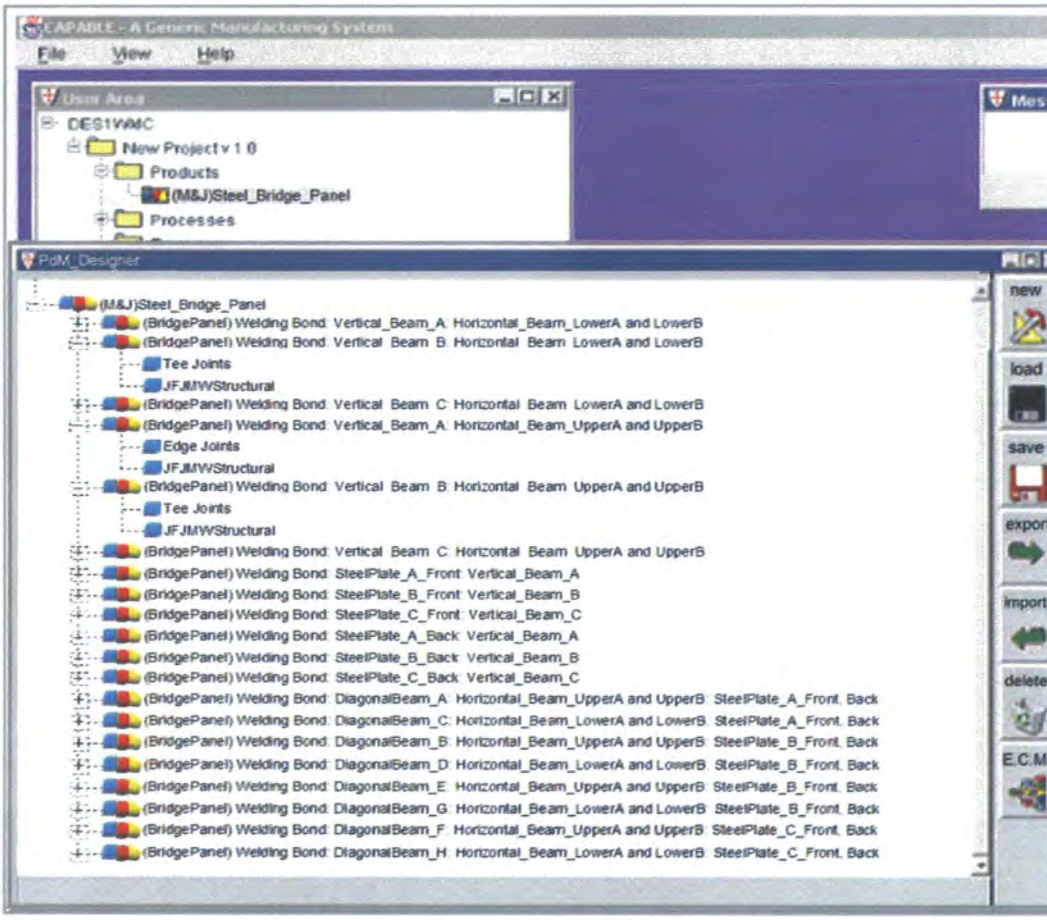


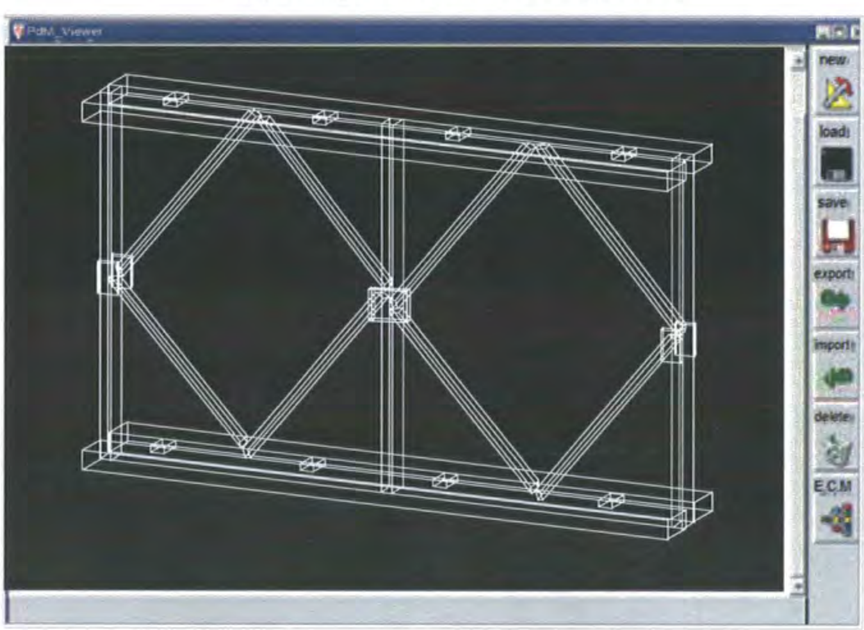
Figure 8.1: Single Steel Bridge Panel

Bridge Items Description	**Material BS standard (steel code)**	**Number Required**	**Weight**
SteelPlate_A_Front and Back = SteelPlate_C_Front and Back	S355	4	1.41 Kg
SteelPlate_B_Front and Back	S355	2	2.82 Kg
Diagonal_Beam_A,B,C,D,E,F,G,H	S355	8	37.90 Kg
Slots	S355	8	1.41 Kg
Vertical_Beam_A=B=C	S355	3	15.50 Kg
Horizontal_Beam_LowerA and LowerB	S355	2	35.48 Kg (in pair)
Horizontal_Beam_UpperA and UpperB	S355	2	35.48 Kg (in pair)

Table 8.1: Bill of Material Description



(a) The product model configuration



(b) A single Bailey's steel bridge panel

Figure 8.2: Example Product Model in the CAPABLE System

8.4.2.2 Factory Model Used in Testing

The blueprint of the factory layout of Mabey & Johnson is shown in Figure 8.3. The layouts are made up by a series of cells and within every cell there is a dedicated workcentre. The type of cells and workcentre(s) are presented in Table 8.2. The table also indicates the type of machines and their operations. However, the table only shows a limited number of the manufacturing operations needed to assemble the steel bridge panel.

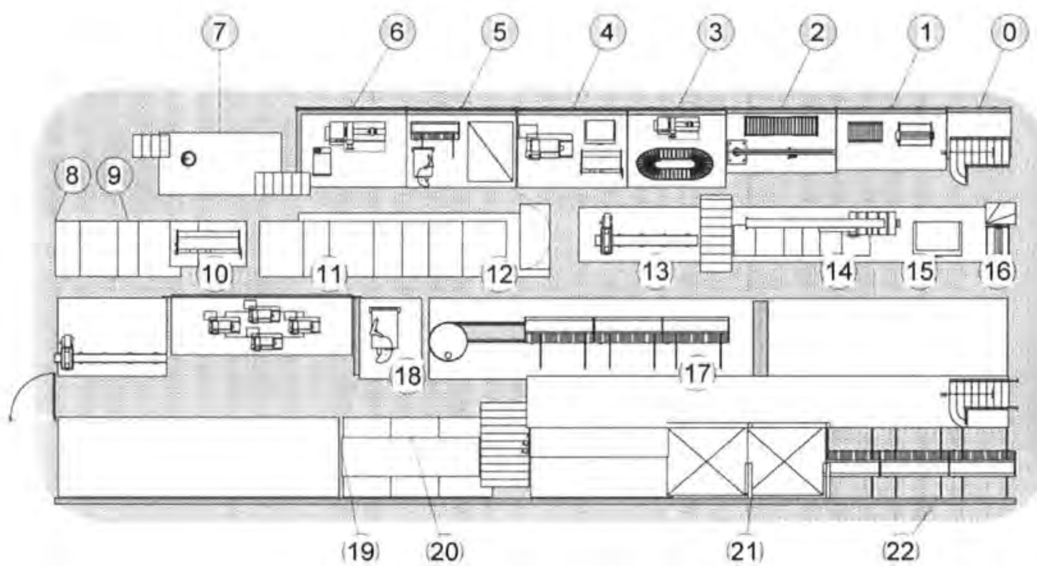


Figure 8.3: Factory Layout of Mabey & Johnson Ltd

Cell Number	Cell Name	WorkCentres (or Machine Types)	Operations
0	Maintenance	Maintenance Area	
1	Robot Cell 01	Robot_Deck_01	
		Robot_Handling (Beams)	Robot_Handling
2	Robot Cell 02	Robot_Deck_02	
3	Robot Cell 03 (Bode)	Robot_Handling (Plates)	Robot_Handling
4	Robot Cell 04	Robot_Weld_Chord_01	
		Robot_Weld_Chord_02	
5	Robot Cell 05	Robot_Arc_Gas_Welder	WP_GasMetalArc_MIG
6	Robot Cell 06	Robot_Arc_Welder	WP_GasMetalArc_MIG
7	Galvanizing Trailer	Surface_Coating	
		Galvanizing	SC_Galvanizing
		Immerse_Washing	SP_Washing
	Inspection Area	Inspection	Inspection
8 & 9	AMT Handling and Drilling	Manual Drilling	MRM_Manual
10	MU Range Panel Drilling	Panel_Drilling_1	MRM_Drilling
		Panel_Drilling_2	
11 & 12	Manual_Weld_Area 01	Manual Weld 01	
		Manual Weld 02	
		Manual Weld 03	
		Manual Weld 04	
13	Pack Saw Area	Pack Saw	
14	Robot Small Parts		
15 &16	Material Storage		
	Weld Cell Transforms		
17	Saw Drill Combind Area 03		
18 - 20	Manual Weld Area 02	Manual Weld	
21	Handling Area		
22	Beam Processing Area		

Table 8.2: Summary of Mabey & Johnson Shop Floor

Figure 8.4 shows the factory design module as modelled in the CAPABLE system. It clearly shows the position of individual cells and associated machine types. Datasheets (see Table 8.3) for robotic handling and welding tools were used to specify process parameters for the robotic centres, giving a range of tools that would be able to perform all handling and welding operations.

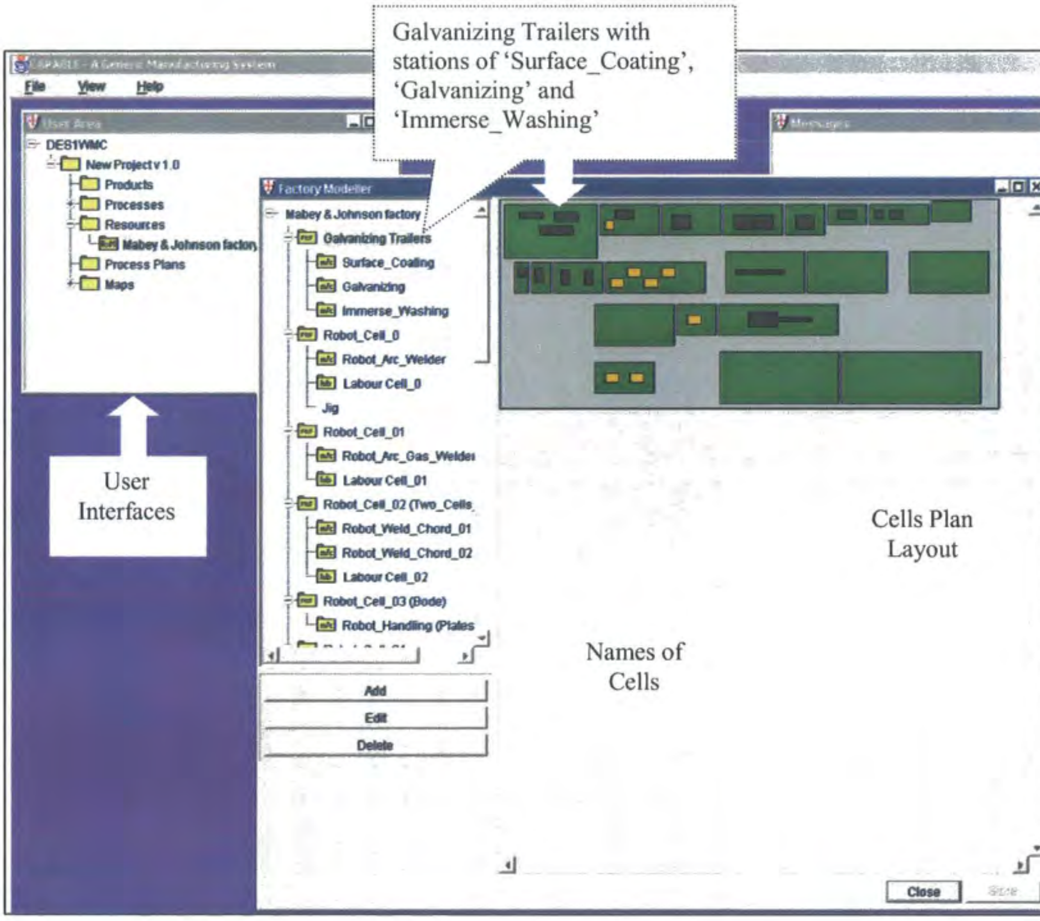


Figure 8.4: Screenshot of the CAPABLE System Factory Model

Resource name	**Max. Travel Speed (mm/s)**	**Max. Weld Flow Rate (mm³/s)**	**Max. Feeding Velocity (mm/s)**	**Max Weld Cord Diameter (mm)**	**Max Loading (kg)**	**Max Arm length (mm)**	**Duration (Hrs)**
 ABB 'IRB 4400'	250	NA	72	NA	60	1950	NA
 ABB 'IRB 1400'	250	NA	72	NA	5	1440	NA
 ABB 'IRB 340'	100	NA	NA	NA	1	NA	NA
 ABB 'IRB 6400PE'	250	3	72	14	120	2500	NA
 ABB 'IRB 6400R'	250	3	72	14	500	2250	NA
 'Galvanising Plant'	NA	NA	NA	NA	NA	NA	24

Table 8.3: Example Resource Model Data

8.4.2.3 Process Model

Process modelling is used to identify the type of process needed to assemble the bridge panel. The Process Model provides specific methods which have been developed to calculate manufacturing time and production quality. Resource Modelling is used to specify process parameters for the selected machining centres i.e. to give a range of tools which will be able to perform all necessary operations associated with the product features. Figure 8.5 shows a user interface of the process model in the CAPABLE system. In particular, it shows different types of welding processes and highlighted a selected process, *WP_GasMetalArc_TIG*. In order to assemble the bridge panel at Mabey & Johnson, other types of processes are also available within the system, these are:

- semi and fully automatic robotic welding,
- manual welding,
- drilling,
- surface coating,
- galvanising,
- immerse washing and inspection.

Furthermore, the company has also imposed Welding Process Specification (WPS), Just-in-time (JIT) and Kanban techniques to improve the shop floor efficiency. Each of the processes and techniques require a high degree of *know-how* to operate successfully.

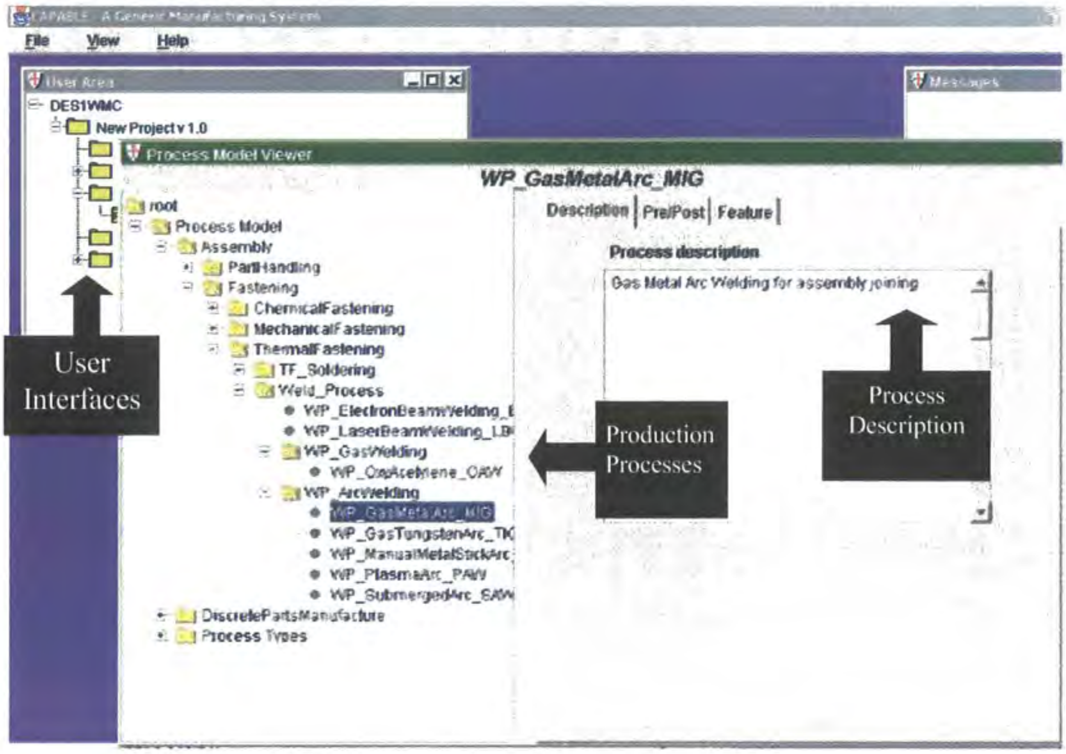


Figure 8.5: Screenshot of the CAPABLE System Process Model

8.4.2.4 Aggregate Process Planning for a Single Steel Bridge Panel

The process plan represents a sequence of assembly and subassembly operations for the construction of a single bridge panel. The steps of the assembly sequence of the final product is made up by a number of subassemblies A, B, C, D, E and F as represented in Figure 8.6. Each of the subassemblies is made up of a number of design feature units as indicated in the diagram. For instance, subassembly A and one unit of vertical beam produced subassembly B. Similarly, subassembly B and one unit of vertical beam produced subassembly C. Once the bill of material for the end product has been prepared, an aggregate process plan can be obtained via the CAPABLE System.

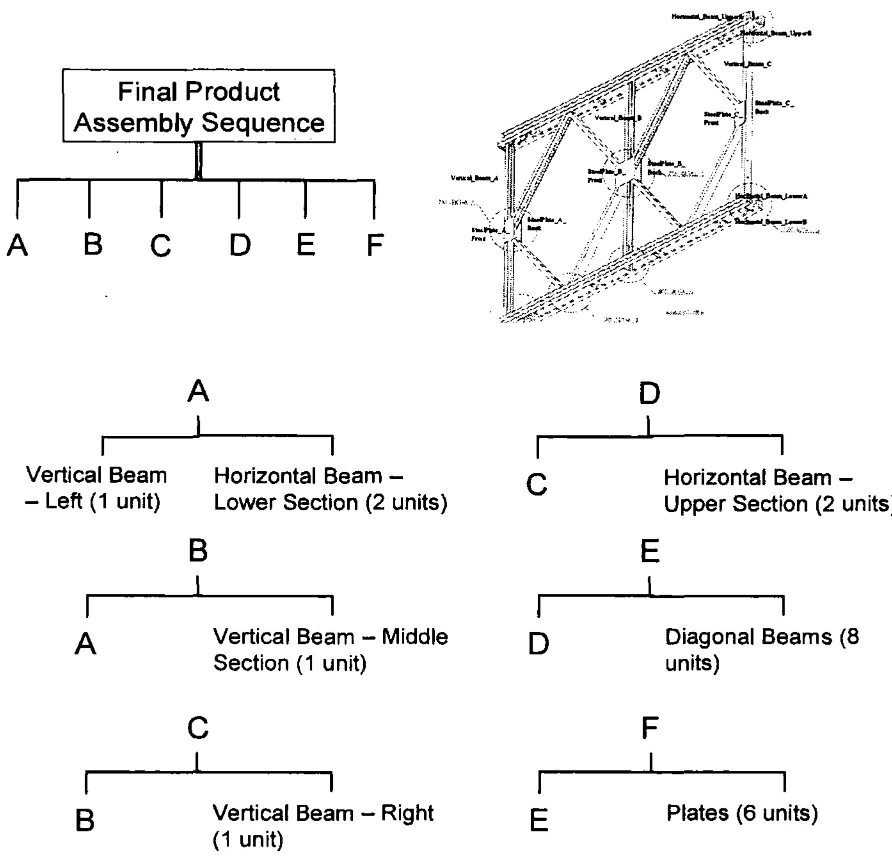


Figure 8.6: Assembly and Subassembly Sequence of a Steel Bridge Panel

8.4.3 Manufacturing Knowledge Acquisition and Reuse

8.4.3.1 Populating Knowledge in the Manufacturing 'Know-how' Knowledge-Based System

The organisational knowledge framework is used to capture manufacturing know-how in the context of product features, processes and factory resources. The manufacturing know-how information is presented in the tables as shown in Appendix F. It is assumed that knowledge acquisition is performed internally by the knowledge experts within Mabey & Johnson Ltd via interviewing and paper based information. The tables in Appendix F show two knowledge types: quantitative and qualitative. Whereas, the knowledge types are further classified into *Written and Benchmarking*, *Observation and Intuition*, *Employee Tacit and Experience* and *Best Practice* as explained in Chapter 6. Figure 8.7, represents the Manufacturing Know-how Knowledge-based system (KBS). The KBS consists of three parts: (1) the main body of the structure, (2) user input dialog and (3) the knowledge instances. The figure illustrates an example of how instances related to a factory are populated and stored in the KBS.

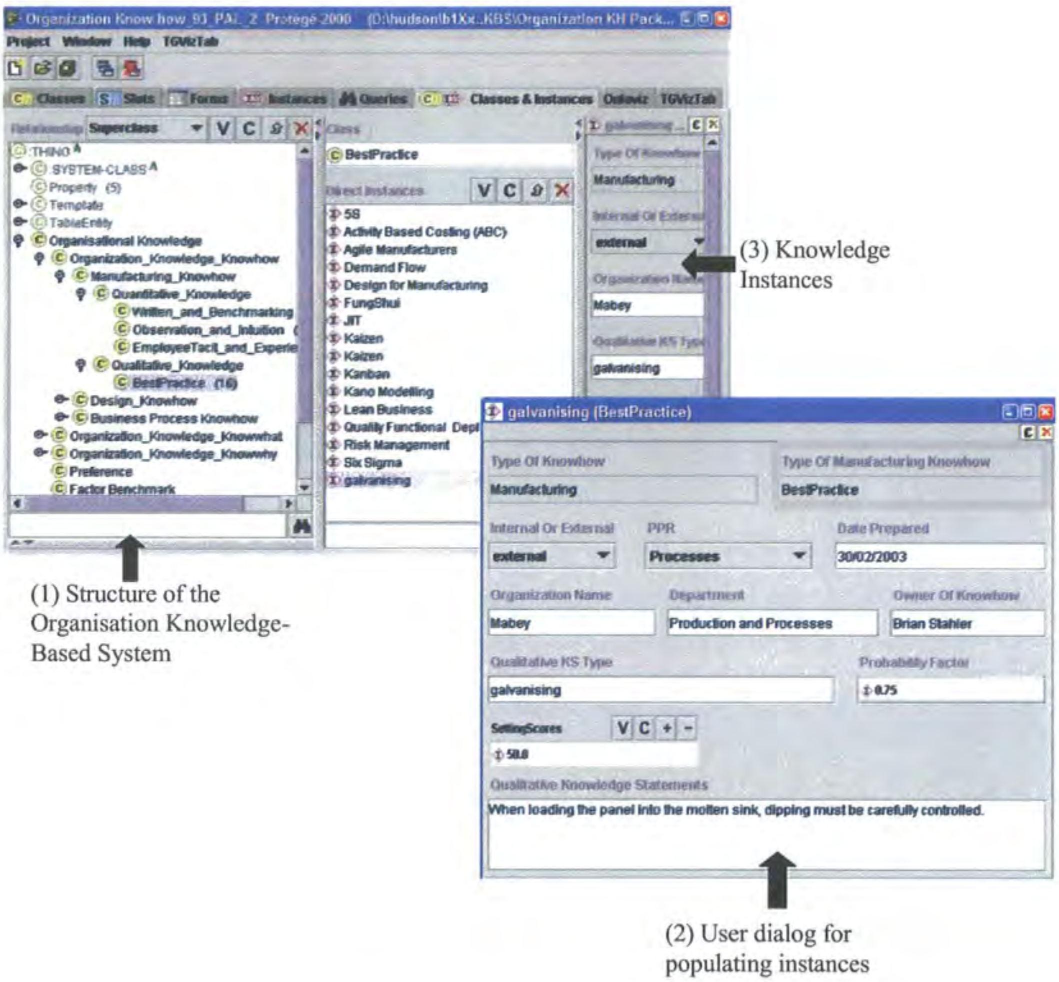


Figure 8.7: Knowledge Acquisition

One of the most important aspects of creating knowledge statements is setting the value of the *Probability Factor* as explained in Section 6.4.3. For instance, *galvanising* is a simple but delicate process and is mostly manually operated which can involve up to six steps to produce the desired coating. Every galvanising '*dip*' takes up to 8 bridge panels at a time. *Dipping speed* varies upon the type of bridge panel and the number of panels going through the molten sink, thus, the *dipping speed* must be carefully controlled to avoid '*air trapped*' when loading the panel into the molten sink (see Figure 8.7). If the *dipping speed* is too high this may create a sudden explosion due to *'air trapped'*. Since *dipping speed* is so important the associated *Probability factor* constraint which defined by axioms is shown in Figure 8.8. Hence when the user chooses the value 'always', axioms tells the system how to interpret the imports numerically. In this case, the highest possible value is '1'. This value will then to be used by the capability analysis method as described in Section 6.3.5. All of these experiences can be stored into the KBS as explicit manufacturing know-how.

```
(exists? Probability
      (and
            (< ('Always'? Probability) 1)
            (< ('Frequently'? Probability) 0.75)
            (= ('Never'? Probability) 0.0)
            (< ('Occasionally'? Probability) 0.25)
            (< ('Sometimes'? Probability) 0.5)
      )
)
```

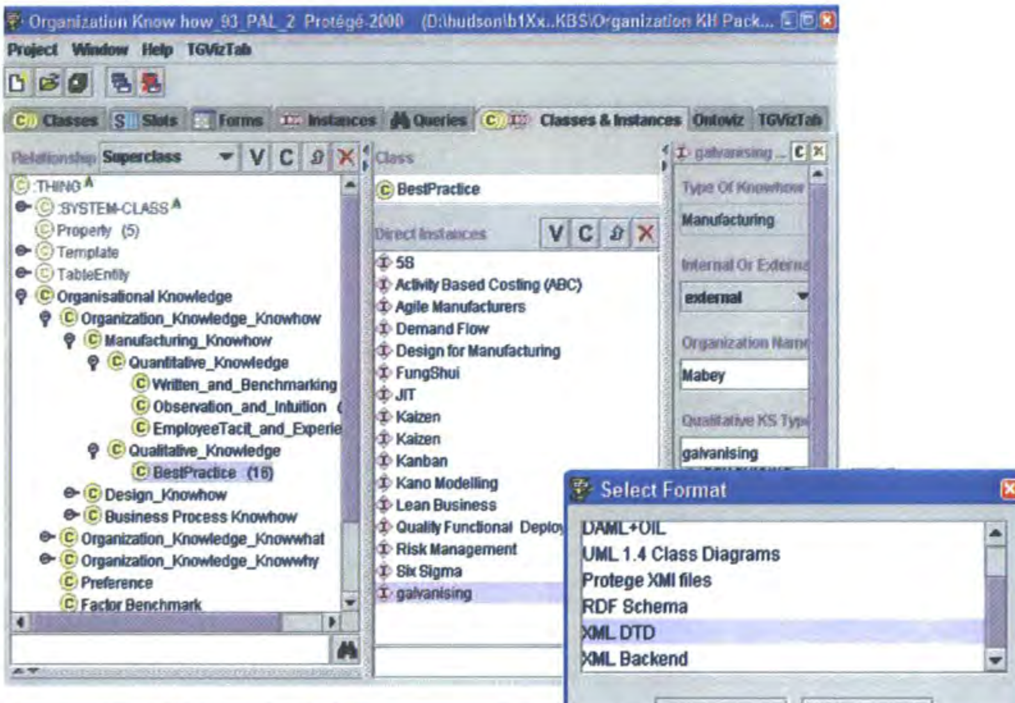
Figure 8.8: Example Probability Factors Defined by Axioms

Another example is the application of *'design_know_how'* within the *organisation_knowledge_know_how*. Design knowledge, covering several processes and resources, was added into the ontology by the knowledge experts. Subsequently, using the CAPABLE system, a process plan was created using feature-to-process and process-to-resource mappings which involved the galvanising process. Taking into account the process parameters (the temperature fluctuation from ambient to 500 °C) the following critical areas of knowledge were immediately identified and presented to the designer:

- Explosions *always* occur if *air trapped* is in the design, for example a closed tube.
- *If Weldments* are present, they should *always* be designed to avoid acid traps
- *Flat panels should normally* be braced to minimise the risk of distortion

These are a few of the examples of using the KBS to capture the expertise to be reused and shared within the product development processes.

Figure 8.9 illustrates an example of how the captured knowledge is being transferred into an XML-formatted file. The next stage of testing involved storing the file into a distributed PDM System and making it ready to be re-used.




```
xml version="1.0" encoding="utf-8" ?>
:!DOCTYPE document (View Source for full doctype...)>
cument>
knowhow>
  <Date_Prepared date="28/02/2003" />
  <SubjectType subject="Galvanising" />
  <FPR>Processes<FPR>
  <Probability_Factor> 0.75<Probability_Factor>
  <Setting_Scores> 50.0< Setting_Scores>
  <Knowledge_Type knowledgetype="Best Practice" />
  <Organization_Name organization="Mabey" />
  <Owner_of_Knowhow ownerknowhow="Brian Stahler" />
  <Qualitative_KS_Type>galvanising</Qualitative_KS_Type>
  <Qualitative_Knowledge_Statements qualitativeKS="When loading the panel into the molten sink, dipping
must be carefully controlled" />
/knowhow>
knowhow>
  <Date_Prepared date="28/02/2003" />
  <SubjectType subject="Robot_Cell_0" />
  <FPR>Processes<FPR>
  <Probability_Factor> 0.75<Probability_Factor>
  <Setting_Scores> 100.0< Setting_Scores>
  <Knowledge_Type knowledgetype="Best Practice" />
  <Organization_Name organization="Mabey" />
  <Owner_of_Knowhow ownerknowhow="Brian Stahler" />
  <Qualitative_KS_Type>Kaizen </Qualitative_KS_Type>
  <Qualitative_Knowledge_Statements qualitativeKS="Means continuing improvement involving everyone and
workers alike" />
<knowhow>
  <Date_Prepared date="28/02/2003" />
  <SubjectType subject="Robot_Cell_0" />
  <FPR>Processes<FPR>
  <Probability_Factor> 0.75<Probability_Factor>
  <Setting_Scores> 100.0< Setting_Scores>
  <Knowledge_Type knowledgetype="Best Practice" />
  <Organization_Name organization="Mabey" />
  <Owner_of_Knowhow ownerknowhow="Brian Stahler" />
  <Qualitative_KS_Type>JIT </Qualitative_KS_Type>
  <Qualitative_Knowledge_Statements qualitativeKS="Just-in-time is a management philosophy that strives to
eliminate sources of manufacturing waste by producing the right part inthe part place at the right time"
/>
/knowhow>
```


Figure 8.9: XML Formatted Knowledge

8.4.3.2 Usage of PDM System to Store Knowledge

Figure 8.10 illustrates a sequence of events of how a XML-based knowledge file can be stored in a PDM system and subsequently downloaded by an external user. The diagram represents:

1. Having created the XML-based knowledge file, the next stage is to invoke the PDM system and use the '*create document*' function to download the knowledge XML-based file into the PDM document storage cabinet.
2. Then to use the XML document by invoking the PDM '*check-out*' function which downloads the document into the user's local file space.

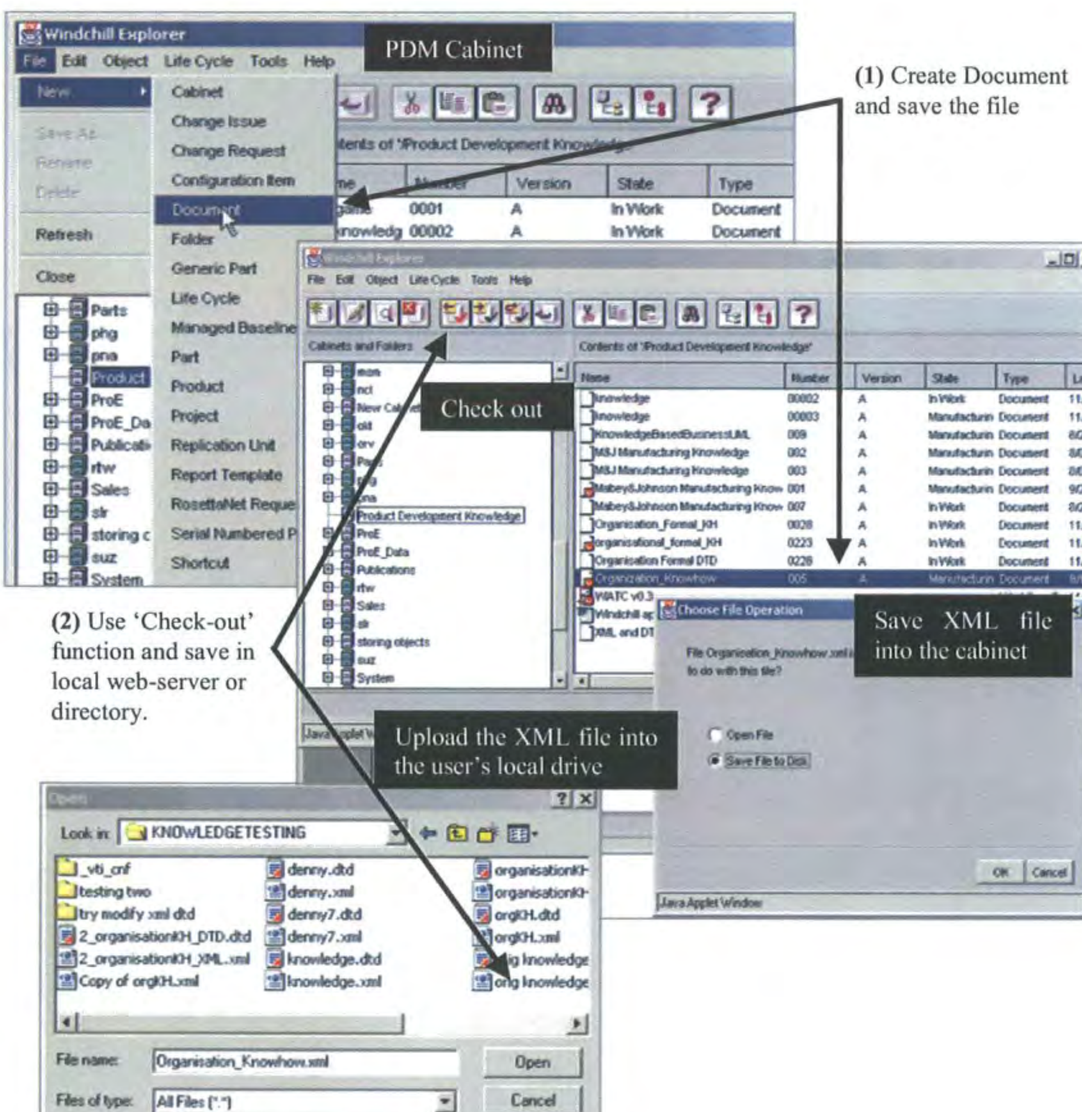


Figure 8.10: Use of PDM in XML-Based Knowledge

8.4.3.3 XML Parser in Knowledge Re-use in Process Planning

This section describes the demonstration of how the application of the XML Parser supports a third party software system. The coordination of the activities is based on the method of WATC. The testing environment is illustrated in Figure 8.11, which depicts an example of Web-based data interoperability between a knowledge-based system, a PDM system, the CAPABLE system and an ERP system. The example shows that the captured knowledge will be saved in a XML file and then placed into a Windchill PDM Cabinet. The diagram also illustrates the links of an XML-based manufacturing knowledge to be re-used by the CAPABLE Aggregate Process Planning system.

The links were established by the XML Parser and defined by *data-string-type* (*extract XML Metadata*). The *data-string-type* is the term used within the XML Parser to identify the *subject-type* in the XML-based knowledge document. Thus, this allows the extracted data to be transferred to the data models in the CAPABLE system. After obtaining an aggregate process plan for the conceptual design with the CAPABLE system, this will then transfer into the PDM system and be ready to be checked-out into the ERP system for capacity planning.

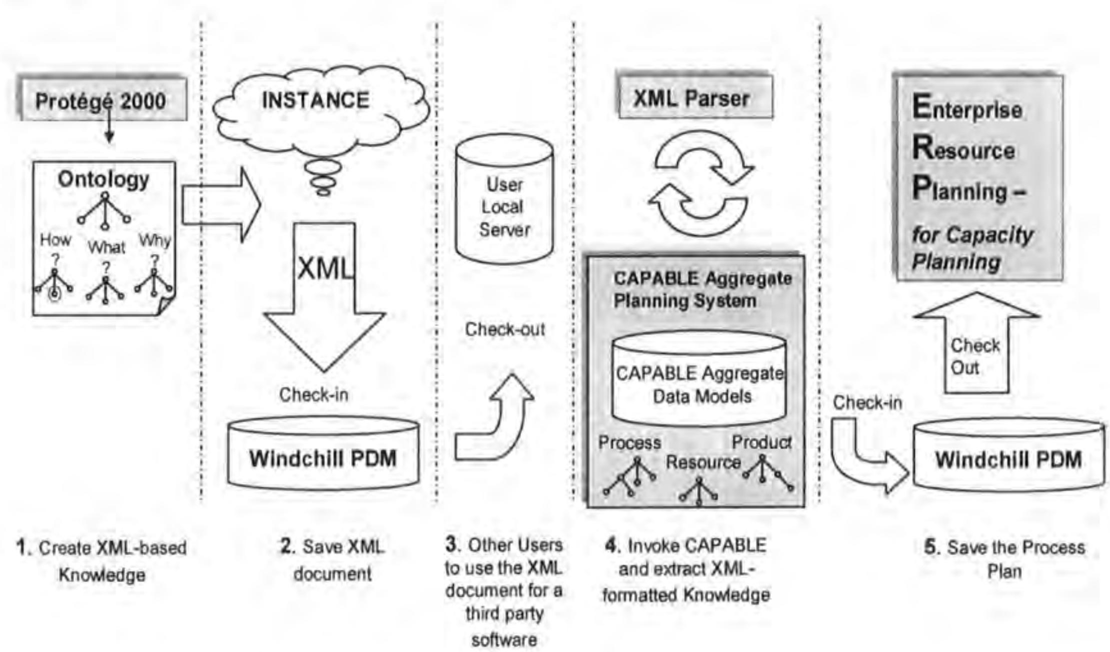


Figure 8.11: Centralised Testing Environment

Figure 8.12 illustrates the knowledge statements related to a specific object termed Robot_Cell_0 which belongs to a station of a factory. As highlighted in the diagram, in order to refine a conceptual design:

1. from the CAPABLE system, a designer will invoke the XML Parser within the data model, for example the factory resource model,
2. the designer will select the XML based knowledge file from the local directory which is already uploaded from the PDM cabinet, and
3. the knowledge statements will then attach to a particular group of machines, which can be used for further analysis to enhance the planning process of a conceptual design. The example shown in Figure 8.12 indicates the resulting knowledge statements extracted to the CAPABLE System which relate to a factory (resource) model object 'Galvanising Trailer' which belongs to a cell of "Mabey and Johnson Ltd" factory.

The design engineer is then required to select the relevant knowledge to refine the design and subsequently run the CAPABLE system to obtain a preliminary process plan based upon these knowledge factors. If the plan requires review, the prioritised knowledge factors obtained from capability analysis, highlight the most appropriate areas of the process plan for improvement based upon the specific instances of knowledge factors used. If the process plan is acceptable, it is then delivered to the PDM system for Plan/Review, and subsequently is readied for implementation in an ERP System for capacity requirements planning.

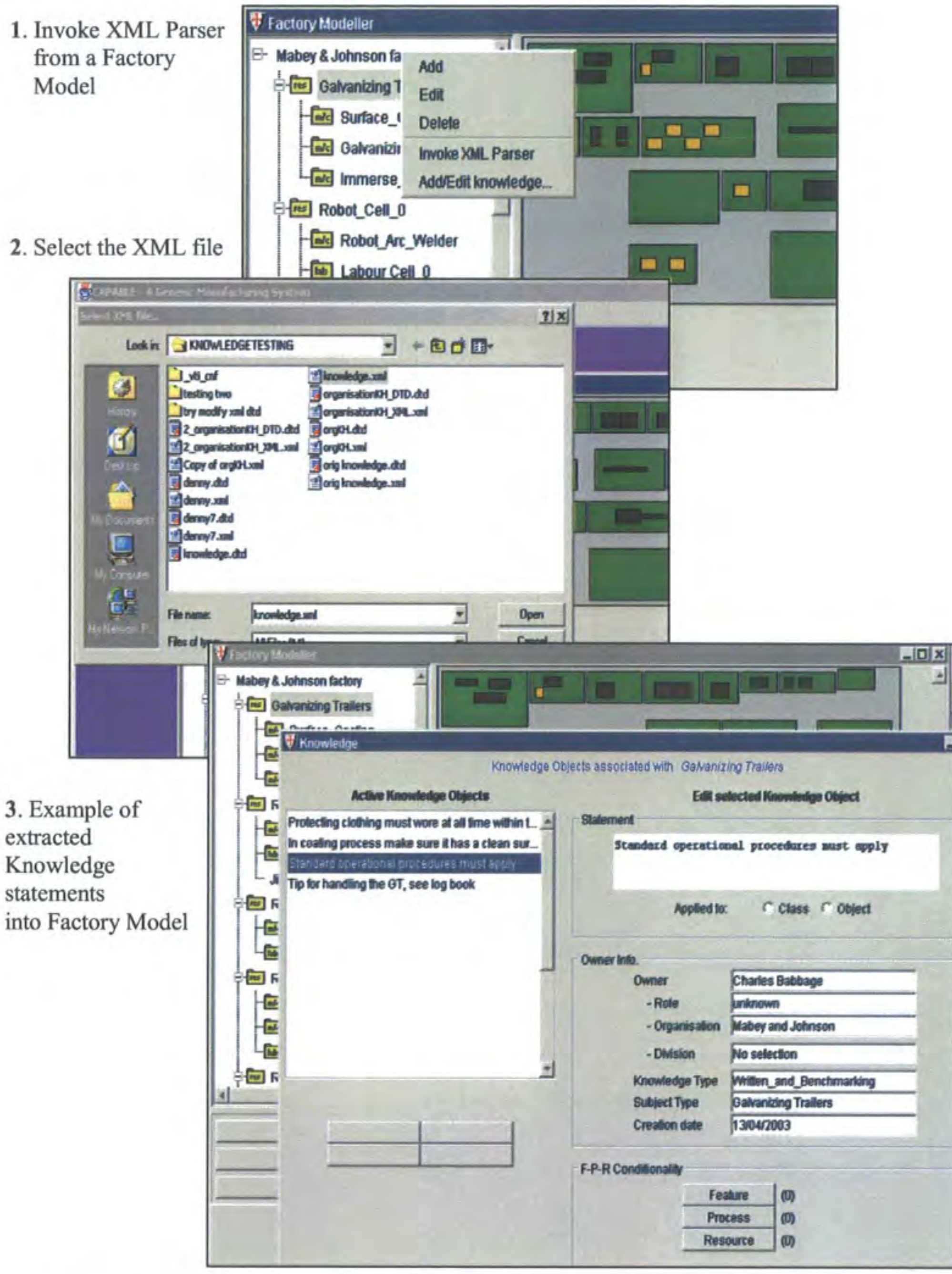


Figure 8.12: Example Knowledge Statements Related to a Specific Object

Figure 8.13 illustrates a process plan generated by the CAPABLE System. However, in order for the ERP system to read and extract the right type of information, the process plan has been converted to a Microsoft Excel file as shown in Figure 8.14. Based on this information the ERP system can estimate the resultant requirements at individual work centres as described in the next section.

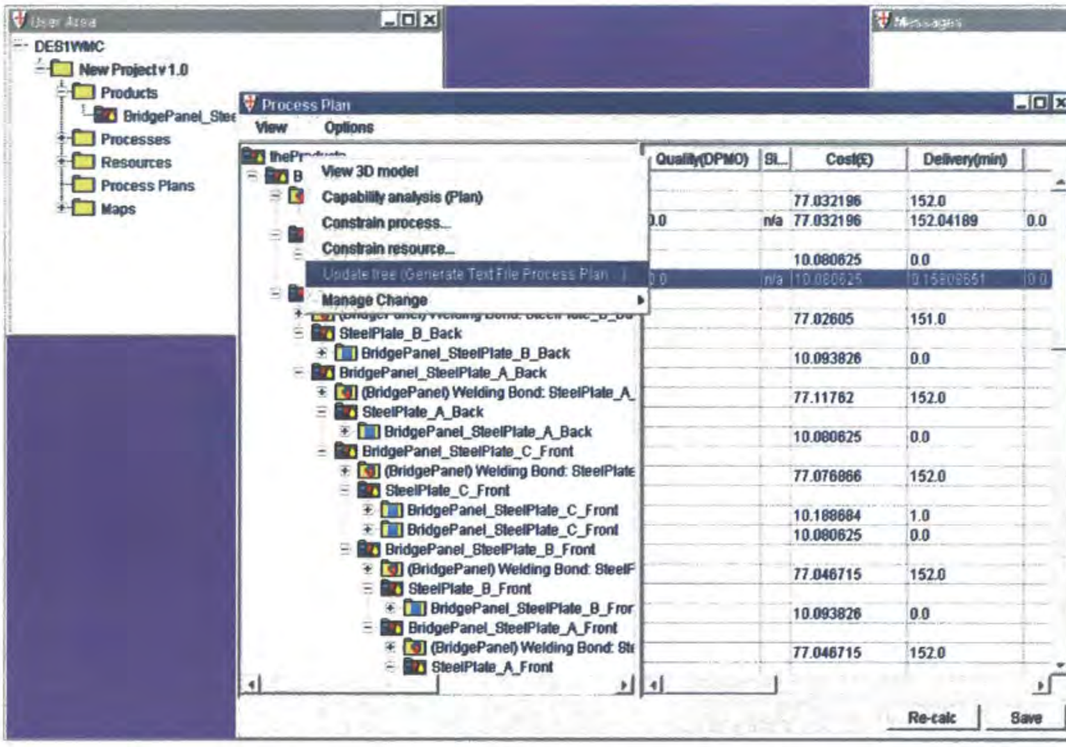
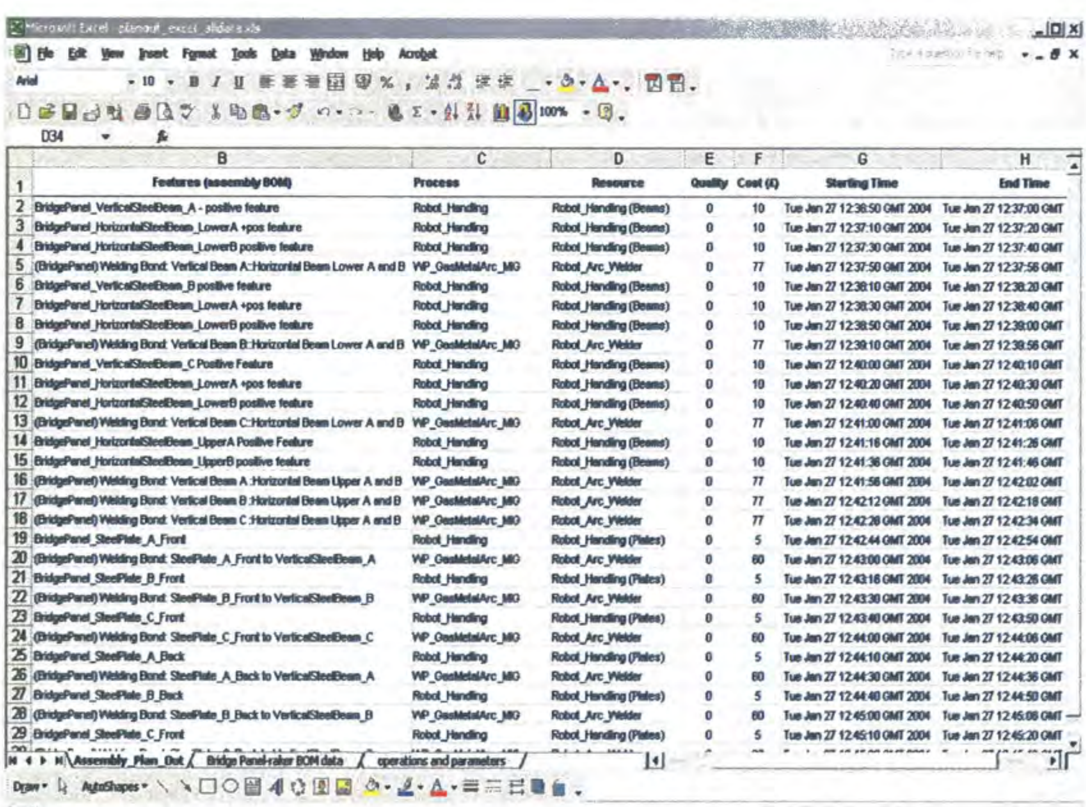


Figure 8.13: Example Process Plan for a Single Steel Bridge Panel



	B	C	D	E	F	G	H
1	Features (assembly BOM)	Process	Resource	Quality	Cost (£)	Starting Time	End Time
2	BridgePanel_VerticalSteelBeam_A - positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:36:50 GMT 2004	Tue Jan 27 12:37:00 GMT
3	BridgePanel_HorizontalSteelBeam_LowerA +pos feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:37:10 GMT 2004	Tue Jan 27 12:37:20 GMT
4	BridgePanel_HorizontalSteelBeam_LowerB positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:37:30 GMT 2004	Tue Jan 27 12:37:40 GMT
5	(BridgePanel) Welding Bond: Vertical Beam A::Horizontal Beam Lower A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:37:50 GMT 2004	Tue Jan 27 12:37:56 GMT
6	BridgePanel_VerticalSteelBeam_B positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:38:10 GMT 2004	Tue Jan 27 12:38:20 GMT
7	BridgePanel_HorizontalSteelBeam_LowerA +pos feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:38:30 GMT 2004	Tue Jan 27 12:38:40 GMT
8	BridgePanel_HorizontalSteelBeam_LowerB positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:38:50 GMT 2004	Tue Jan 27 12:38:00 GMT
9	(BridgePanel) Welding Bond: Vertical Beam B::Horizontal Beam Lower A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:39:10 GMT 2004	Tue Jan 27 12:39:56 GMT
10	BridgePanel_VerticalSteelBeam_C Positive Feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:40:00 GMT 2004	Tue Jan 27 12:40:10 GMT
11	BridgePanel_HorizontalSteelBeam_LowerA +pos feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:40:20 GMT 2004	Tue Jan 27 12:40:30 GMT
12	BridgePanel_HorizontalSteelBeam_LowerB positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:40:40 GMT 2004	Tue Jan 27 12:40:50 GMT
13	(BridgePanel) Welding Bond: Vertical Beam C::Horizontal Beam Lower A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:41:00 GMT 2004	Tue Jan 27 12:41:06 GMT
14	BridgePanel_HorizontalSteelBeam_UpperA Positive Feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:41:16 GMT 2004	Tue Jan 27 12:41:26 GMT
15	BridgePanel_HorizontalSteelBeam_UpperB positive feature	Robot_Handing	Robot_Handing (Beams)	0	10	Tue Jan 27 12:41:36 GMT 2004	Tue Jan 27 12:41:46 GMT
16	(BridgePanel) Welding Bond: Vertical Beam A :Horizontal Beam Upper A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:41:56 GMT 2004	Tue Jan 27 12:42:02 GMT
17	(BridgePanel) Welding Bond: Vertical Beam B :Horizontal Beam Upper A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:42:12 GMT 2004	Tue Jan 27 12:42:18 GMT
18	(BridgePanel) Welding Bond: Vertical Beam C :Horizontal Beam Upper A and B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	77	Tue Jan 27 12:42:28 GMT 2004	Tue Jan 27 12:42:34 GMT
19	BridgePanel_SteelPlate_A_Front	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:42:44 GMT 2004	Tue Jan 27 12:42:54 GMT
20	(BridgePanel) Welding Bond: SteelPlate_A_Front to VerticalSteelBeam_A	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	60	Tue Jan 27 12:43:00 GMT 2004	Tue Jan 27 12:43:06 GMT
21	BridgePanel_SteelPlate_B_Front	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:43:16 GMT 2004	Tue Jan 27 12:43:26 GMT
22	(BridgePanel) Welding Bond: SteelPlate_B_Front to VerticalSteelBeam_B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	60	Tue Jan 27 12:43:30 GMT 2004	Tue Jan 27 12:43:36 GMT
23	BridgePanel_SteelPlate_C_Front	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:43:40 GMT 2004	Tue Jan 27 12:43:50 GMT
24	(BridgePanel) Welding Bond: SteelPlate_C_Front to VerticalSteelBeam_C	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	60	Tue Jan 27 12:44:00 GMT 2004	Tue Jan 27 12:44:06 GMT
25	BridgePanel_SteelPlate_A_Back	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:44:10 GMT 2004	Tue Jan 27 12:44:20 GMT
26	(BridgePanel) Welding Bond: SteelPlate_A_Back to VerticalSteelBeam_A	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	60	Tue Jan 27 12:44:30 GMT 2004	Tue Jan 27 12:44:36 GMT
27	BridgePanel_SteelPlate_B_Back	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:44:40 GMT 2004	Tue Jan 27 12:44:50 GMT
28	(BridgePanel) Welding Bond: SteelPlate_B_Back to VerticalSteelBeam_B	WP_GasMetalArc_MIG	Robot_Arc_Welder	0	60	Tue Jan 27 12:45:00 GMT 2004	Tue Jan 27 12:45:06 GMT
29	BridgePanel_SteelPlate_C_Front	Robot_Handing	Robot_Handing (Plates)	0	5	Tue Jan 27 12:45:10 GMT 2004	Tue Jan 27 12:45:20 GMT

Figure 8.14: Example Process Plan in MS Excel Format

8.4.4 ERP for Capacity Requirements Planning and Testing Results

The ERP system used in this case study is called Compiere. Since the system has many different functions available, the ERP system must be customized. Selecting the appropriate functions can reduce the implications of using the system. The procedural steps of customizing the system are as follows:

1. Define the main menu with the functions as shown in Figure 8.15. For example, the correspondence functions which uses in the implementation are:
    a. Import File Loader
    b. Resource
    c. BOM drop
    d. Product
    e. Production
    f. Asset

Resource function captures the relevant resource data, such as the machine and station types. *Product* function captures the requirement of product features. *BOM Drop* captures the assembly and process sequences. *Production* captures information related to the *Client (customer)* such as description of the product, production plan, line and date of movement. *Asset* is the final function used to capture detail views of the *resource* requirements such as the availability, location and delivery.

2. Import the CSV file using the *Import File Loader* function

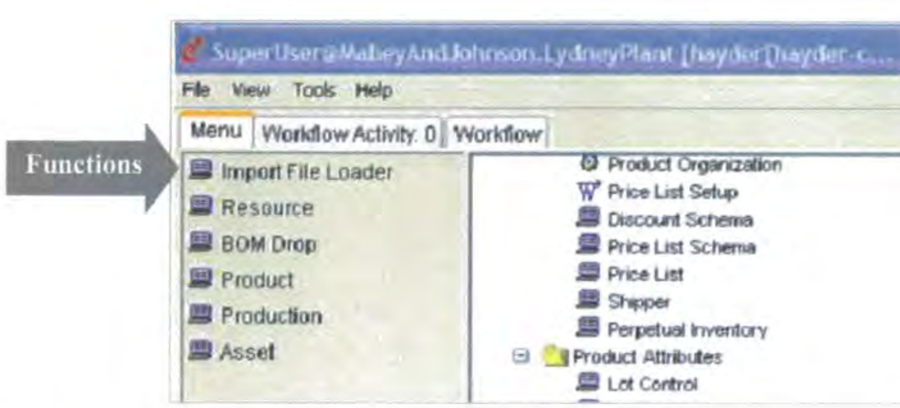


Figure 8.15: Compiere Functional Requirements for Capacity Planning

3. Fill in all additional information, for example, in the *product* function as illustrated in Figure 8.16. Information such as weight, height of the product feature, cost and so on.

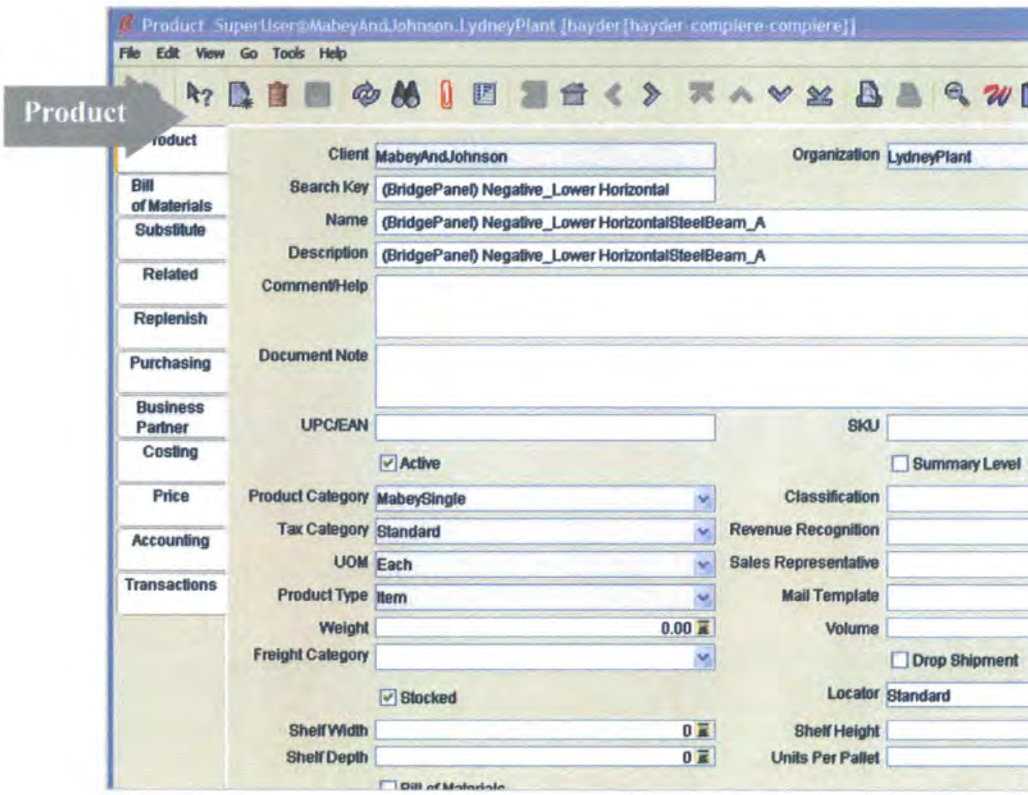


Figure 8.16: *Product* Function

The illustration shown in Table 8.4 indicates the Capacity Requirements Planning generated by the Compiere system. The table illustrates the impact of the time-phased capacity information. The total workload of 1050 hours and its percentage allocations to each workstation over a 10 day period for 50 panels per day. One of the reasons to use a 10 day period is to show the periodic changes according to the manufacturer's Master Production Schedule (MPS) requiremnets.

The timing of workcentres varies, for instance, for the workstation *Galvanizing* at time period 2, a capacity requirement was planned for 16.5 hours and *Immerse_Washing* was planned for 7.5 hours. This indicates that the total processing time for the 50 panels in *Galvanising* is 16.5 hours and 7.5 hours in *Immerse_Washing* respectively. Further particulars the table has indicated is the concern of long range capacity planning, thus, the system is able to generate and provide indications of capacity planning requirements for a longer period of time.

	Time Period											
Workstations	1	2	3	4	5	6	7	8	9	10	Total Hours	Workstations Percentage (%)
Robot_Deck_01	0	0	0	0	0	0	0	0	0	0	0	0
Robot_Handling (Beams)	12	12	12	9.5	9.5	17	17	17	18.5	18.5	143	13.6
Robot_Deck_02	0	0	0	0	0	0	0	0	0	0	0	0
Robot_Handling (Plates)	6	6	6	9	9	9	9.5	9.5	9.5	9.5	83	7.9
Robot_Weld_Chord_01	0	0	0	0	0	0	0	0	0	0	0	0.0
Robot_Weld_Chord_02	0	0	0	0	0	0	0	0	0	0	0	0.0
Robot_Arc_Gas_Welder	16	16	16	17.5	17.5	17.5	17	17	20	20	174.5	16.6
Robot_Arc_Welder	16	16	16	17.5	17.5	17.5	17	17	20	20	174.5	16.6
Surface_Coating	7	7	7	7.5	7.5	7.5	8	8	9	9	77.5	7.4
Galvanizing	16.5	16.5	16.5	17.5	17.5	17.5	17	17	20	20	176	16.8
Immerse_Washing	7.5	7.5	7.5	8.5	8.5	8.5	9	9	10	10	86	8.2
Inspection	5	5	5	7.5	7.5	7.5	8.5	8.5	9.5	9.5	73.5	7.0
Manual Drilling	0	0	0	0	0	0	0	0	0	0	0	0.0
Panel_Drilling_1	5	5	5	6	6	6	7	7	7.5	7.5	62	5.9
Manual Weld 01	0	0	0	0	0	0	0	0	0	0	0	0
Total	91	91	91	101	101	108	110	110	124	124	1050	100

Table 8.4: CRP for a Full Panel

8.4.5 Closing Remarks of Case Study 1

The case study was used to illustrate the concept of bridging the discontinuity in communicating early design concept and manufacturability evaluations using a centralised network configuration (Section 8.4). The case study has demonstrated how the use of PDM technology can enhance a distributed and collaborative environment to support the product development process (Section 8.4.3.2).

The case study has also demonstrated how manufacturing knowledge can be represented using an ontological approach and be reused by an aggregate process planning system through the application of an automatic data exchange mechanism (Section 8.4.3). In particular, the case study has demonstrated its performance of supporting the designers to refine a design at the conceptual stage by adding manufacturing knowledge into the CAPABLE system to generate alternative early process plan then use it in an ERP system to obtain and optimised a rough-cut capacity planning (Section 8.4.4).

Within Mabey & Johnson Ltd, all technical knowledge is tacit and possessed by experts. However, the methodology demonstrated in this research was used to capture, store and re-use this knowledge within the process of collaborative product development. The case study has demonstrated that:

1. Early collaboration in product design using the WATC methodology can maximise the opportunity for optimising designs.
2. With the increasing of knowledge *know-how* in product design and manufacturing capabilities, the application of organisational knowledge-based system coupled with the aggregate manufacturing modelling method can be used to capture and maximise the amount of available information when designing customised products.
3. The integration environment using enterprise technologies can enhanced the speed of feedback and used to support decision making and enable the design to be right first time.

How much lead time can be reduced on new product introduction is difficult to measure at this stage as the methodology only tested the early stage of product development processes. Since the knowledge-based system has been created using an ontological approach, the terms and meanings defined could be further developed into an industrial standard which would make the derived *know-how* applicable across entire production networks.

8.5 CASE STUDY 2: ARVINMERITOR VEHICLE-DOOR-LATCH

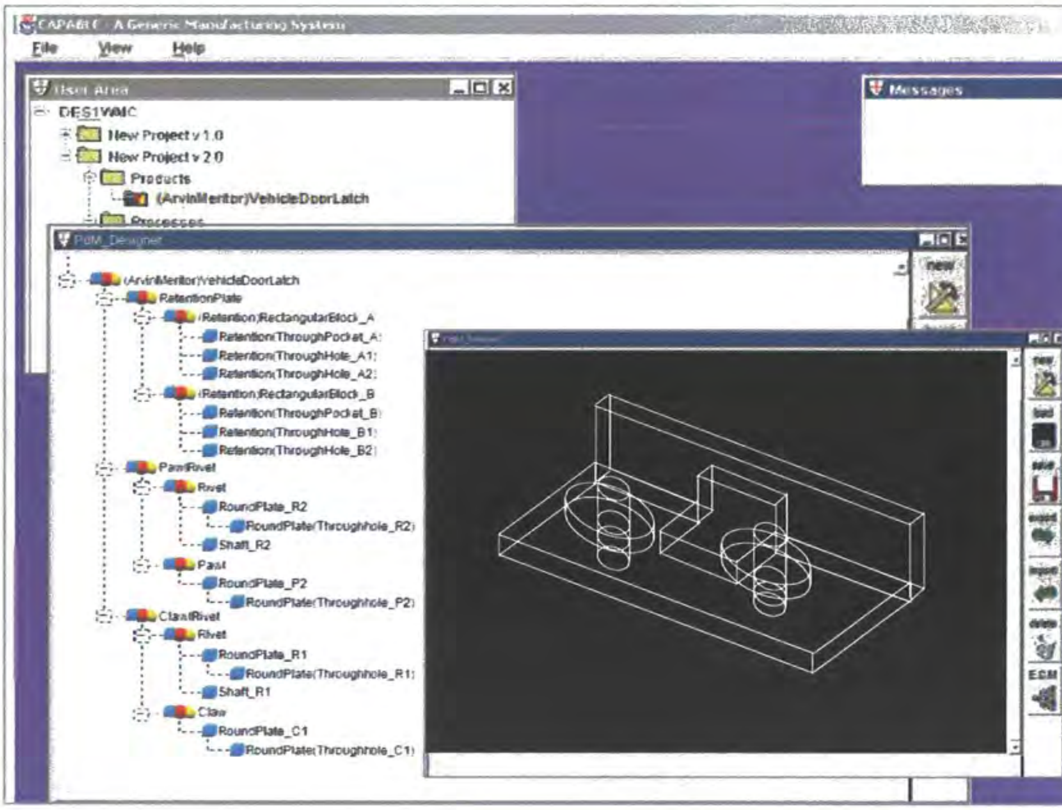
8.5.1 Objectives and Aims

ArvinMeritor Automotive is a large Tier-1 automotive supplier. The investigation of their concept design and development processes was carried out with the Access Control Systems division based in Birmingham. The products of this division are security latches and powered latches for commercial vehicles and cars. The company designs, produces prototypes and assembles the latches. The actual component production is carried out by sub-contractors. The tooling for the latch assembly is supplied by a subcontractor, however the 'jig' is designed in-house. In terms of the manufacturing environment, this industrial collaborator is quite different from Mabey & Johnson. *ArvinMeritor* is a global enterprise with different subsidiaries all over the world including the US, UK, mainland Europe and China. However, as the research progressed, the majority of their manufacturing activities were slowly being outsourced to their Far East and mainland Europe subsidiaries, thus, the plans of the case study had to be changed. The case study was carried out at their UK branch in Birmingham and with assistance from their customers and suppliers.

As described previously, Case Study 1 was based on a centralised collaborative production network which focused on the applications of the CAPABLE system, a Knowledge-Based System, PDM and ERP. In contrast, this case study reports on the combination of a de-centralised and a centralised network in collaborative product development and knowledge management platform for ArvinMeritor and their customers and suppliers in real-time. The overall aim of Case Study 2 is to prove that the use of P2P technology is more economically feasible and enables the controlled sharing of their knowledge with smaller enterprises to form a collaborative network with larger companies as introduced in Section 7.4.2.

8.5.2 Modelling of a Vehicle-Door-Latch within the CAPABLE

There are over eighty different components in the selected vehicle-door-latch. However, only a few core components have been selected to demonstrate the research hypothesis at the conceptual design stage which is shown in Figure 8.17. Figure 8.17 (a) illustrates the EBOM and the conceptual design stage of the product which has been modelled within the CAPABLE system. Figure 8.17 (b) illustrates a detailed representation of the selected components. This provides a simplified design without compromising the features of the support tool.



(a) The conceptual design configurations of a vehicle-door-latch

(b) Detail design of a selected Vehicle-Door-Latch component

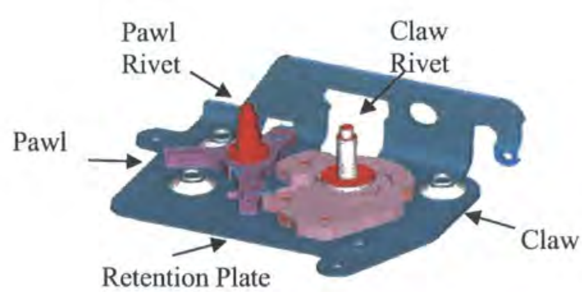


Figure 8.17: Product Model of a Selected Vehicle-Door-Latch Component

It was assumed that the general assembly of a single vehicle-door-latch was made up of five individual components: the retention plate, pawl, pawl rivet, claw and claw rivet. The way to assemble the latch was used the *orientation* class in the assembly process model performed by *feeding equipments* in the resource model. As mentioned in Case Study 1, Process Model was used to identify the type of process applicable in assembling the vehicle-door-latch. To assemble the vehicle-door-latch the Process Model provides specific methods which have been developed to calculate manufacturing time and production quality. The Resource Model was used to specify process parameters for the selected machining centres i.e. to give a range of tools able to perform all necessary operations associated with product features.

8.5.3 Product Development Knowledge Acquisitions and Re-use

The types of knowledge captured in the organisational knowledge-based system for this case study was classified into customer requirement definition, concept definition and engineering product/project knowledge of which were separated into Phases 0, 1, 2, 3, 4 and 5 as shown in Table 8.5.

Phase	Knowledge Types
0	customer/sales/marketing related
1	concept design
2	engineering
3	testing
4	manufacture
5	decommissioning

Table 8.5: Product Development Knowledge Representation in ArvinMeritor Automotive

This case study was interested in Phase 0 and Phase 1 primarily i.e. to capture design and customer knowledge and also the interaction with the preceding and following phases. Phase 0 turns an enquiry into a profitable order. There are many reasons to reject enquiries and bad business should be eliminated. Secondly, bespoke knowledge is required to be stored, for example, to convert customer information into customer knowledge.

In the conceptual design stage, Phase 1, the design engineers concede that a large amount of useful concept and engineering ideas and historical information is lost because there is no specific way of entering the information, for instance, *lessons learned, best practice* and *benchmarking* during the concept design. At ArvinMeritor there were some attempts to capture and store these types of knowledge using document-based and spreadsheet solutions. In addition, the design concept generated tends to be a generic type of design rather than a specific configuration for a particular vehicle model. For example, in the conceptual design stages there are different types of Nissan vehicle-door-latches, the designs tend to be a global family type of products rather than a specific configuration for a particular vehicle model. As such the aim of the knowledge acquisition was to classify and transform the knowledge types into a more generic term.

8.5.4 Utilisations of a Centralised and a De-centralised Networks in Product Development

It has been recognised that current PLM implementations are document oriented, have a non-customisable data model and encounter inter-enterprise integration difficulties. To overcome these problems a hybrid integration methodology using P2P and PLM technologies has been developed. The overall diagram of P2P and the Product Development Integration Architecture is shown in Chapter 7.

Figure 8.18 shows the high-level view of the open source server and user interface architecture. At the lowest layer, it consists of the database. This can be chosen from the range of relational databases available, and for this test case SAP-DB was utilised due to its stability, ease of use and close integration with the PHP (Personal Home Page) scripting language. On top of this layer are the two open source gateways; one is an Apache web server for serving static and PHP based web pages and the middle tier is an the Tomcat servlet engine which serves the Java server pages (JSP) based applications. These three layers (database, web server and application server) form the server side of the system.

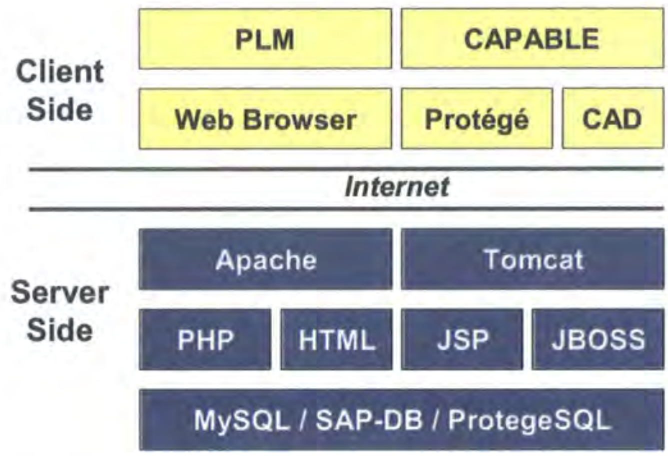


Figure 8.18: High-Level View of Open Source Server and User Interface Architecture

On the client side, the user has four main applications:

1. The web browser through which interactions with the PLM system are carried out.
2. The Protégé Java applet that allows the user to query and manage the knowledge base and.
3. A CAD system to enable the user to create and manipulate the STEP based models held in the PLM system.
4. The CAPABLE system for process planning

8.5.5 Application of Peer-to-Peer in Collaborative Environment

The implementation was kept as simple as possible. As mentioned previously, the Protégé2000 ontology development and user interface environment was used. The back end consists of the open source SAP-DB database with the Java database connectivity (JDBC) connector to Protégé2000. Connectivity is achieved using an open source implementation of JXTA open standard P2P network protocol (www.jxta.org). Implementation of JXTA is in Java 2 standard edition, an extension to enable RDF queries and ontologies to be shared over P2P is used to share the Protégé knowledge base. Figure 8.19 illustrates the topology of the system. In this instance the architecture was setup with a super-peer net in order to create a number of virtual servers at each of the project participant's sites. These super peers are visible as peers to both the other super-peers and the internal peer net which they serve. They also act as the default rendezvous peer, and peer information is shared between the super-peers in order to improve the redundancy and query speed of the system. The systems' settings enable enterprises and users without static addresses to collaborate using dynamic addressing, as well as the users being able to work offline. Thus, it empowers users in all possible circumstances. The grey area contains the internal peer-net of the enterprise with the interconnected PLM and knowledge-based system servers and workstations. Outside the peer-net is the external peer-to-peer environment which interlinks all the super-peers from every collaborator together. This enables peers to connect to each other and query and manipulate knowledge with other peers on the internal and wider peer-net transparently.

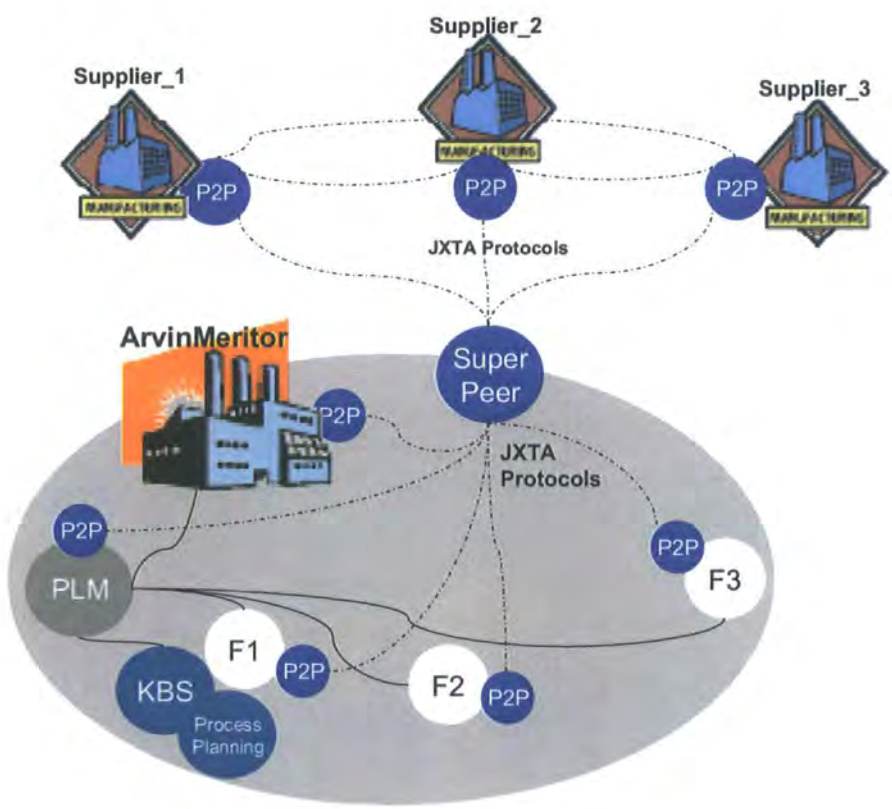


Figure 8.19: Super Peer Architecture Showing the Interacting within Internal and External Peer Nets of a Collaborative Project.

8.5.6 Testing and Results of Using Peer to Peer in Collaborative Environment

The results shown in Table 8.6 were tested empirically using the data and a small inter-enterprise setup. The server was located in Cranfield, and users concurrently created the new project, generated new concepts and optimised the design gradually from two remote locations to simulate the collaborative scenario as shown in Figure 8.18. All ontologies deployed shared a common back end. The applications were tested on identical hardware with identical configurations and internet connections. The apparatus were Dell Xeon workstations with 1GB, RAM 15K rpm disks and 1 Mb internet connection. The operating system was Microsoft Windows 2000 on all machines. The reason to use 1 Mb internet connection was to minimize the cost of bandwidth, this was one of the advantages of decentralised network application. The test suffices for a small scale pilot. The test simulated a small scale virtual enterprise (VE) setup through from initiating the VE to running the system within the collaborative network. This setup has a typical VE with a large number of small servers distributed within a common network, a small number of transactions are processed at each node. Table 8.6 compares the key features for a collaborative project environment, these were:

1. **Ontology:** how easy was it to enter, query, and reuse both the ontology and domain knowledge within the collaborative group? Since this is common to all the tools, the only difference was the way in which the ontology was stored within the different systems.
2. **VE setup:** how rapidly could the VE be set up and start to operate on all the remote sites?
3. **Deployment time:** The time to customise and deploy the entire project including the project management framework and communications setup.
4. **Lifecycle Management:** The ability to control the state of a document, manage versioning and history of the data.
5. **Total Cost of Ownership:** The costs of licences, implementation, system integration, training and maintenance.

Application	ontology	VE setup	Deployment time	Lifecycle management	Total cost of ownership
Windchill	RDFs ontology in document container, access via Windchill	B2B integration via Info*engine (one-to-many integration), manual	1 month for small project.	Graphical workflow	Server & client licence, implementation and maintenance
Pure P2P	RDFs ontology in java application	Peer to peer, automatic discovery	1 day	Version control and rule based system	Training on P2P and knowledge application
SuperPeer net	RDFs ontology in java application	SuperPeer automatic discovery	1 day	Version control and rule based system	Training on P2P and knowledge application

Table 8.6: Comparison of Key Features of Tested System

8.5.7 Closing Remarks of Case Study 2

The case study was used to illustrate the concept of a hybrid integration methodology using P2P and PLM technologies. The case study has demonstrated how the use of client/server systems with the independence and interoperability of P2P systems can enhance the effectiveness of collaboration and data distribution to support an inter-enterprise setup (Section 8.5.5). In particular, the case study has demonstrated the comparison of using the traditional client/server PLM and P2P. The result of the comparison has shown in Table 8.6.

The application of the combination using centralised and decentralised methodologies fulfilled the following requirements:

- Satisfied the use of domain specific ontologies distributed to the clients (users) via the P2P network and open standards: Elimination of the interoperability issues for product and project knowledge and easier set up for inter-enterprise collaboration.
- Fulfilled through the use of RDF ontologies that can be queried by agent based systems. Efficient query and retrieval mechanisms.

- The third requirement is met by de-centralisation: Elimination of centralised bottlenecks in bandwidth and resources, empowerment of collaborators within networks to "control" the knowledge they create.
- In addition the methodology described, open source can eliminate software licence costs, a solution to the problem of vendor lock-in in the long term, elimination of unnecessary complexity and freedom to modify the application. Platform and application independence: Enable the enterprise to concentrate on its work and not be tied in to any vendor.

8.6 ANALYSIS OF THE CASE STUDIES

The fundamental hypothesis and logic that underpin the product integration architecture and a hybrid integration methodology have all been validated (Sections 8.4.3.3 and 8.5.5). The experimental results show that:

1. The centralised product development integration architecture can support the communication and translate the design concept and manufacturing scenarios and the support of knowledge management method to refine design decisions (Section 8.4.3.2).
2. The support of WATC method is able to fulfil and demonstrate a collaborative environment from customers requested to refine the design concept (Section 8.4.3.3).
3. The application of the organisational knowledge-based system captures tacit knowledge and is able to support the early design decision (Section 8.4.3.1).
4. However, PDM systems are inherently centralised, overly complex to set up and require a long period of time for customisation. As such a hybrid integration methodology has been fulfilled to prove the usefulness of the application of P2P (Section 8.5.5).
5. The hybrid integration method can fulfil the autonomy of smaller enterprises to create and control their knowledge which would not be possible using the centralised integration architecture (Section 8.5.6).

8.7 SUMMARY

One important aspect is that this research was focused on building a prototype software environment. The principal objective was to demonstrate the use of internet-based communication tools and Web-based technologies to enhance and bridge the gap of communicating in early product requirements as effectively as possible.

As previously described in Chapter 2, different paradigms have been proposed to support distributed network environments. All those paradigms however, were developed during the mid 1990s to early 2000s. Since then Web-based technologies, enterprise integration systems and data exchange formats have evolved. As such, this research has proposed and contributed a different approach to deploy an 'out-of-box' solution with the utilisation of centralized and de-centralized production networks and the integration of disparate software systems. It has been proved that the product development integration architecture provides an alternative framework for using Web-based technologies in capturing manufacturing knowledge together with enterprise integration systems and data exchange format to support other software components in collaborative product developing environments.

Case study 1 used a Bailey's bridge panel as an example to describe the testing of the overall product development integration architecture and concluded the advantages of the applications of Web-based technologies, enterprise integration systems in a distributed and collaborative product development environment. Furthermore, Case Study 1 described the use of a process plan based on the assembly sequence of a single Bailey bridge panel. The application of an open source ERP system was used for generating a periodical table for capacity requirements planning. Case study 2 used an automotive door-latch as an example to describe the testing of the combination of decentralised and centralised production network using the P2P technologies. The study created a virtual enterprise collaboration to compare the applications of centralised PDM, pure P2P and SuperPeer.

A new project is under development in open source PLM (Salustri 2006). The Opensource PLM (OPLM) project aims to provide SMEs with the ability to perform the needs of current proprietary PLM functions, but, freely available without the prohibitive resource requirements of the larger systems. In the future version of the integration architecture, further investigation will be carried out to integrate with this new open source PLM when it is available.

Chapter 9

DISCUSSION, RECOMMENDATIONS AND CONCLUSIONS

9.1 INTRODUCTION

In response to the research hypothesis of bridging the discontinuity of early design decisions, this research has developed a product development integration architecture to address a collaborative product development and distributed manufacturing knowledge environment (Section 7.2). In particular, the support was concerned with design concepts and manufacturing performance evaluations from the critical early stages of product development throughout the product life cycle. The research sets out to support the internet-based process of communicating new product requirements using PDM, ERP, the development of a data coordination and exchange mechanism (Section 7.3), knowledge-based systems (Section 6.3) and an aggregate process planning system (Section 5.3). In addition, the research has also explored the applications of hybrid P2P networking in a decentralised production network environment (Section 7.4).

The main focus of this chapter is to discuss the limitations of the techniques leading up to the construction of the experimental system and its performance. These include the deployment of Protégé2000 and the development issues using the ontological approach in manufacturing knowledge 'know-how' (Section 6.3.2). In addition, the discussions will directly address the applications of the PDM system to store XML based knowledge to be reused as well as issues in relation to the XML Parser (Section 7.3.2.1). An overall discussion outlines the benefits and the practicalities of the overall system are also highlighted. Recommendations for process planning and knowledge management tools are also suggested accordingly. Further recommendations of future works to enhance the product development integration architecture are also suggested. Finally this chapter will conclude with a brief review of the overall research.

9.2 DISCUSSION

9.2.1 Existing Work

In response to the main objectives of the research (Section 1.3.2):

- A comprehensive review of published literature has been presented (Chapter 2).
- An assessment of PDM systems and an industrial survey of using PDM have been carried out (Chapter 3).
- The main topics of the investigation and implementation were discussed (Chapter 5, 6, 7 and 8).

Furthermore, managing manufacturing knowledge in a collaborative and distributive product development environment was also identified as a key element of this work, together with the support of enterprise integration systems. In particular, this research work proposed the application of enterprise integration systems and Web-based technologies which can bridge the discontinuity of communication within the design and product development teams. It has also been stated that much of this work relates to the management of capturing and representing manufacturing knowledge. The effectiveness of managing design and manufacturing knowledge was also a vital ingredient for enhancing intuitive collaboration across the distributed enterprise.

9.2.2 Specific Contributions to the Research

The contributions of the research are to enhance the decision support and manufacturing evaluation throughout the enterprise, these include:

- The translation of product specification into process requirements and the creation of an organisation knowledge framework (Section 5.2 and Section 6.3).

- The theoretical definition and formalisation of the concept of a distributed and collaborative product development and manufacturing knowledge management was presented and realised (Section 7.1).
- Creating new methods of linking design and manufacturing for the integration of a collaborative product development to enhance early design communication (Section 7.3).

This was achieved by developing and making enhancements to the following key technology components:

1. The concept of an aggregate product model was realised and enhanced via the creation of specific capability in welding (Section 5.3.1). A library of aggregate process models has been expanded to cover Bailey's steel bridge panels and ArvinMeritor's vehicle-door-latch assemblies (Section 5.3.2). The aggregate resource model has been enhanced for the representation of the manufacturing capability of companies in a distributed enterprise (Section 5.3.3).
2. The theoretical aspects and applications of a novel methodology for exploiting a knowledge management editor tool that structures organisational knowledge has been defined and developed using an ontological approach (Section 6.3). This approach can be used to elaborate organisational knowledge by defining the semantics to capture the meaning of the terms and axioms (to define a set of rules if applicable) and to enhance and encapsulate the way of reusing the knowledge-based system in a collaborative manner within a production network (Section 6.4.3).
3. A product development integration architecture has been defined and developed by linking design and manufacturing domains using Web-based and enterprise integration technologies (Section 7.2). The significance of using a XML Parser to support the systems interoperability and data exchange has also been implemented. A novel "Workflow Activity Task Controller" (WATC) methodology has been defined to implement the *time based integration wrapper* concept in the interactions between generic types of PDM, ERP, knowledge-based systems and an aggregate process planning system (Section 7.3).
4. A hybrid P2P framework, which is a subset of the product development integration architecture, has been defined and developed. This part of the work was initiated by Cranfield, and the author has contributed to several elements of

the design such as indexing, security issues, the use of workflows and refining the connectivity of the infrastructure (Section 7.4). The aim of using P2P is to demonstrate the use of a decentralised network for product development collaboration between smaller companies and larger corporations. The framework also shows a degree of flexibility that can be adapted into different network configurations (Section 7.4.3).

9.2.3 Data Collection

Meetings with the collaborating companies established that the results were credible. Acquiring the right kind of data can enhance the accuracy of the experimental system. The use of technical terms in the work is important too as this can avoid the confusion of the terms being used in the knowledge-based system. There were two types of data to be collected from the industrial collaborators. Firstly, the manufacturability of processes, factory resource and product design information for the CAPABLE System. Secondly, qualitative and quantitative knowledge via interviews with product development experts and paper based information.

The format of data for the product model was stated as design features. The information was presented in the form of an aggregate product model, describing the product model as a combination of features for a specific type of product. For example, the work described in this research mainly dealt with Bailey's steel bridge panels and ArvinMeritor's vehicle-door-latch. The data required for the Process model and the Factory Resource models were well defined and developed by the CAPABLE/Space project. However, the intention of this work was to develop the CAPABLE system to enable the system to read and extract XML information as described in Section 5.5 and Section 7.3.2.1. Further development was to introduce a welding domain into the aggregate product model, additional assembly processes for the aggregate process model and factory modelling of plan layouts (Section 5.3).

The information of the additional data for welding and assembly processes were based on the specification acquired from the industrial collaborators. For instance, the entities being used to describe the welding processes such as speed of the robotic movement, travelling distance, temperature, size of the welding cord and so on. The data for this work however, was mostly related to the product development stages from concept to embodiment.

Data acquired for the knowledge-based system was more complicated and sensitive. Two of the major issues were 'trust' and 'willingness'. Trust was an issue as the companies may be wary to disclose sensitive information to store into a system to be shared among other companies. Therefore, the experts had to be willing to share their knowledge and expertise. However, the research assumed that those issues were resolved. Knowledge acquired for the system was two-fold i.e. qualitative and quantitative knowledge, which is known as 'know-how' related to product design and manufacturing at the shop floor level (Section 6.3.1).

In product design, a specific example is knowledge in tolerance studies, where the working conditions of the relationship of parts are vital. For instance, in designing a Bailey's steel bridge panel, issues such as temperature range for expansion allowances, loading, span-of-bridge and stress factors for design characteristics in structure safety and yield. Another type of design knowledge is bespoke services, for example, customisable products or a one-off product requests. The common points of customised product requirements to be considered within Mabey & Johnson Ltd are material and reinforced structure, for example, material characteristics of steel plate and cost justification. Within manufacturing tolerance, issues of material tolerance and material selection form the basic fundamentals of designing a product. The allowance of mechanical tolerance in machining such as Saw, Pack, and Cutting processes must also be considered.

In terms of designing a product that requires design features with access and drainage involves the galvanising process, the benchmarking of good design for filling, venting and drainage requires:

- Means for the access and drainage of molten zinc.
- Means for escape of gases from internal compartments (venting).

Manufacturing knowledge on the shop floor is all about knowledge of experience, techniques and best practice related to production processes and factory resources as explained in Chapter 5. For instance, both of the industrial collaborators deployed manufacturing best practice within the shop floor with such practices as Just-in-Time

(JIT) and Kanban. Other knowledge can be captured, such as experiences of using equipment and safety procedures.

9.2.4 Overall System Evaluation

The product development integration architecture has been developed to demonstrate how manufacturing data can be used collaboratively and distributed using the latest PLM technology. The in-house aggregate process planning system, CAPABLE, has been further developed to capture data related to the collaborators' product design, process capabilities and factory resources.

The process planning system demonstrated the capability of streamlining a company's product design at a conceptual level. However, the CAPABLE System was originally built as a research tool. Further development is needed in order for it to be used as a commercially applicable software system. The application of a Product Data Management System (PTC Windchill) proved that it is very effective in terms of sharing data and documents-based information in a collaborate and distributive environment. In addition to its *versioning* control capability, the system enables the user to locate the latest changes to relevant product information and documents. This is particularly applicable to companies with complex product ranges as they require effective control and collaborations with their vendors. The drag and drop user-friendly workflow function makes the implementation of a company's workflow processes quick and easy. For companies such as ArvinMeritor, this technology offers the flexibility to accommodate their collaborative environment.

During the course of the research, it has been established that the use of a centralised PDM system would be more beneficial if it is diverted to an alternative approach in sharing information with smaller companies. The main benefits of this novel approach can provide the autonomy of sharing data, bandwidth and reduce the cost of deploying high-end hardware and large scale software systems. As a result of this research, a P2P networking approach seems to be the future strategy for information sharing and collaboration between larger corporation and smaller companies. An additional point that needs to be emphasized is the application of the PDM's workflow function. A workflow created using the PDM system can only be used within that particular PDM native environment. The workflow cannot be reused by a different PDM system. This is inflexible in terms of adopting the applications within a collaborative and distributive environment.

In addition, a XML Parser data exchange mechanism has been implemented to convert knowledge statements to support process planning in product design decision making. This methodology has a great deal of benefits if a company is looking for new technology and a methodology to adopt the concept of enterprise integration. The limitation of this XML Parser however, is that it is specifically built for exchange of XML-formatted information for the CAPABLE system. Further enhancement is needed in order to interoperate with other third party software.

Compiere CMPCS covers all manufacturing activity within the various types of production environments. The system is not completely developed yet, and there is no 'out of the box integration' with other enterprise systems requiring specific application development to fit it into an existing enterprise environment. The system performance over low speed dial-up connections is as yet very poor (however the web interface is usable). It also cannot be deployed with another open source database.

Further software systems have also been evaluated for the development of an ontological-based knowledge system. The software packages are OntoEdit and SemanticMiner (www.ontoprise.com). Unlike Protégé2000, OntoEdit and SemanticMiner are more user friendly, they are not just built for computer scientists but also for engineers who may require some basic training to understand the concept of ontology and knowledge-based systems. The software allows integrated inferencing capabilities, drag and drop semantic mapping, spread sheet import for Microsoft Excel, as well as database import for SQL, Oracle and DB2. The software is currently used by companies such as BMW, Cathay Pacific and several smaller American airlines. This software would be beneficial for companies who are considering building an in-house knowledge-based system to streamline and support their product design and manufacturing efficiency as well as to maintain their greatest asset, knowledge.

9.3 RECOMMENDATIONS AND FUTURE WORK

There are several areas that are deemed to be in need of further investigation. The first concerns the application of the data exchange mechanism in the integration architecture. The XML Parser is especially built for the CAPABLE system. Thus, a generic XML Parser should be built as a stand-alone mechanism that provides the functionality and capabilities of interoperability with third party software system. The RDF parser is designed to run in a Web-browser which can be used to extract information on both XML and RDF based documents on the client side. However, as far as the author is aware the support of the RDF parser in the WWW community is not fully completed yet. As for future work, the research should investigate and upgrade the application of the RDF parser into the product development integration architecture.

In terms of XML documents, future work should develop a *middle-tier* software using XML DML (Data Modification Language) technology to enable users to update the XML document. XML DML stores XML data in a typed form directly in SQL statements into a database which creates indices on XML data type columns and improves database performance. As for the organisational knowledge-based system, future work is planned to define and embed the next layer of the ontology in "*Know-what*" and "*Know-why*".

There are a few open source ERP systems available in the current open source *community*, these include ERP5 (www.erp5.org), Opentaps (www.opentaps.org/index.php), Fisterra (www.fisterra.org) and CK-ERP (www.freshmeat.net/projects/ck-erp/). A new Opensource PLM (OPLM) (www.oplm.org/) is also available. In the future version of the integration architecture, further investigation should be carried out to implement with these new open source systems.

9.4 CONCLUSIONS

In this research work, the dissertation has been focused on and paid special considerations to the proposal of the integration architecture to support collaborative product development and knowledge distribution in manufacturing industry. The implementation of these solutions in cooperation with Web-based and various software supporting systems has been analysed and discussed. The implementation and testing case studies focused on the early stages of product development. However, the infrastructure of the integration architecture is also suitable for the latter stages of product development providing that suitable software systems are used which can read and extract XML based information.

With the advancement and capabilities of Web-based technologies, the integration architecture is also geographically independent and can be used collaboratively and remotely. The significance of using XML-formatted data to support systems interoperability and knowledge re-use has been demonstrated. It was also found that there is no single management tool or data exchange format that can satisfy all requirements and overcome all the obstacles involved within collaborative product developing environments. However, the proper application of Web-based technologies, PLM technologies and other proprietary tools are a means of bridging the gap between the designs and manufacturing departments for improving the product development process.

As a result of the theoretical and practical experiences in using ontology and knowledge management, the research has specified and created an Organisational Knowledge-Based System compatible with aggregate process planning. With the development of user-friendly ontology editing software and automatic data exchange functions, the application of ontological approaches to exchange information across the WWW is most likely to be an essential aspect of the next generation of global knowledge management tools. However, there are still issues that need to be addressed in order for the approach to help and be accepted by the company. For instance; are the knowledge experts willing to share their expertise? Is the cooperative knowledge safe to disclose? How is the knowledge being validated?

Finally, the new P2P framework methodology as a subset of the product development integration architecture is a suitable solution for collaborating enterprises especially SMEs. The framework can create, manage and reuse their knowledge, as well as collaborate within Virtual Enterprises easily and without expense. In addition, the combination of a de-centralised network and client/server is possible through the application of distributed workflows implementation and enterprise integration systems' native workflow function.

As more and more *open source* solutions emerge the *open source community* have learned that this rapid evolutionary process presents a more flexible approach than the traditional closed model.

REFERENCES

1. **Abecker, A., Bernardi, A., Maus, H., Sintek, M. and Wenzel, C., (2000),** "Information Supply for Business Processes: Coupling Workflow with Document Analysis and Information Retrieval", *Journal of Knowledge-Based Systems*, **13**, pp.271-284.

2. **Aberdeen Group, (1999),** "Collaborative Product Commerce: Delivering Product Innovations at Internet Speed", *Market Viewpoint*, **12**(9).

3. **Agility Group, (2004),** Transferring Knowledge and Technology to Business, http://www.dur.ac.uk/agility/page.php?pageID=1
(Site Visited December 2004)

4. **Al-Hawamdeh, S., (2002),** "Knowledge Management: Re-thinking Information Management and Facing the Challenge of Managing Tacit Knowledge", *Information Research*, **8**(1), ISSN 1368-1613.

5. **Allen, D. K. and Alting L., (1986),** "Manufacturing Processes: Student Manual. Utah", USA: CAM Software Research Laboratory.

6. **Applegate Directory, (2002),** http://www.applegate2.co.uk/all/index.htm.
(Site Visited June 2002)

7. **Aziz, H., Gao, J.X., Maropoulos, P.G. and Cheung, W.M., (2005),** "Open Standard, Open Source and Peer to Peer tools and Methods for Collaborative Product Development", *Journal of Computers in Industry*, **56**, pp. 260-271.

8. **Aziz, H., Gao, J.X., Maropoulos, P.G. and Cheung, W.M., (2004),** "A Design Environment for Product Knowledge Management and Data Exchange", *Methods and Tools for Co-operative and Integrated Design*, (ed.) S. Tichkiewith and D. Brissaud, Kluwer Academic Publishers, ISBN 1-4020-1889-4, pp.257-267.

9. **Baker, R.P., and Maropoulos, P.G., (1998),** "Manufacturing Capability Measurement for Cellular Manufacturing Systems", *International Journal of Production Research,* **36** (9), pp.2511-2527.

10. **Becker, J., zur Muehlen, M. and Marc, G., (2002),** "Workflow Application Architectures: Classification and Characteristics of Workflow-based Information Systems", Workflow Handbook 2002. Future Strategies, Lighthouse Point, FL 2002, pp. 39-50.

11. **Beckman, T.J., (1999),** "The Current State of Knowledge Management", In Knowledge Management Handbook, Edited by J. Liebowitz, pp.1.1-1.22, NY: CRC Press.

12. **Beizer, M., (2002),** "Interesting Times for Workflow Technology", extracted from "The Workflow Handbook 2002", published in association with the Workflow Management Coalition (WfMC), 428 pages, ISBN 0-9703509-2-9.

13. **Betteridge, M.J., (2000),** "A Methodology for Aggregate Assembly Modelling and Planning", PhD Thesis, University of Durham, School of Engineering, Durham, UK, 2000.

14. **Bond, J., (2001),** "Business Uses of Peer to Peer (P2P) Technologies, A Netmarkets Europe White Paper", Netmarkets Europe.

15. **Boudreau, M.C. and Robey, D., (2000),** "Organizational Transition to Enterprise Resource Planning Systems: Theoretical Choices for Process Research", *Proceedings of the 21st International Conference on Information Systems,* Brisbane, Australia, 20-13.12.00, pp.291-299.

16. **Bourke, R. W., (2001),** "Best Practices in Web-Enabled Collaborative Product Development", *2PDM e-zine*, **4**(2).

17. **Bouthillier, F. and Shearer, K., (2002),** "Understanding Knowledge Management and Information Management: the Need for an Empirical Perspective", *Information Research,* **8**(1).

18. **Bramall, D.G., (2006),** "Analysis of Manufacturing Operations using Knowledge-Enriched Process Planning", PhD Thesis, University of Durham, School of Engineering, Durham, UK.

19. **Bramall, D.G., Colquhoun, P.G., McKay, K.R. and Maropoulos, P.G., (2001a),** "A System Architecture for Distributed Aggregate Process Planning", *Proceedings of the 17th National Conference for Manufacturing Research,* IMechE, UK, pp. 269-274.

20. **Bramall, D.G, McKay, K.R., Colquhoun, P.G. and Maropoulos, P.G., (2001b),** "Supporting Aggregate Process Planning with Product, Process and Resource Knowledge", *Proceedings of 8th European Conference on Concurrent Engineering (ECEC'2001),* Universidad Politecnica de Valencia, Spain, pp.17-21.

21. **Bramall, D.G., McKay, K.R., Maropoulos, P.G., Rogers, B. and Chapman, P., (2002),** "A Capability Analysis Method for the Technical Assessment of Qualitative Design and Process Planning Knowledge", *Proceedings of the 9th European Concurrent Engineering Conference,* Modena, Italy, pp.15-17.

22. **Bramall, D.G., McKay, K.R., Rogers, B.C., Chapman, P., Cheung, W.M. and Maropoulos P.G., (2003),** "Manufacturability Analysis of Early Product Designs", *International Journal of Computer Integrated Manufacturing,* **16**(7-8), pp.501-508.

23. **Brown, J.S. and Duguid, P., (1998),** "Organizing Knowledge", *California Management Review,* **40**(3), pp.90-111.

24. **Browne, J., Harhen, J. and Shivnan, M., (1988),** Production Management Systems: A CIM Perspective, Addison Wesley, England, ISBN 0-201-17820-6.

25. **Burdick, D., (1998),** "PDM 1997 Year in Review", Research Note- CIM, M-03-5341, Gartner Group.

26. **Burgess, T.F., (2001),** A General Introduction to the Design of Questionnaires for Survey Research, Information Systems Services, University of Leeds, Edition 1.1.
http://www.leeds.ac.uk/iss/documentation/top/top2.pdf.
(Site visited December 2006)

27. **Burkett, M., Jemmeter, J. and O'Marah, K., (2002),** "Product Lifecycle Management: What's Real Now", AMR Research Report, (http://www.amrresearch.com/)
(Sites visited August 2004)

28. **Burkett, M., O'Marah, K. and Carrillo, L., (2003),** "CAD versus ERP versus PDM: How best to Anchor a PLM strategy", AMR Research Report, http://www.ptc.com/WCMS/files/17636en_file1.pdf.
http://www.amrresearch.com/Content/View.asp?pmillid=16264.
(Sites visited August 2004)

29. **Cheung, S.C., Chiu, K.W. and Till, S., (2003),** "Data-driven Methodology to Extending Workflows to E-services over the Internet", *Proceedings of the 36th Hawaii International Conference on System Sciences,* ISBN: 0-7695-1874-5.

30. **Cheung, W.M., Maropoulos, P.G., Gao, J.X. and Aziz, H., (2004),** "A Framework for Distributed Workflows, Peer-to-Peer and PLM/PDM Collaborations to support OEMs and SMEs", *CD-Rom Proceedings of the 2nd International CIRP Seminar on Digital Enterprise Technology (DET04),* Seattle, Washington USA, September 13 – 15.

31. **Choo, C.W., (2006),** The Knowing Organization: How Organizations use Information to Construct Meaning, Create Knowledge, and Make Decisions, 2nd Edition, New York: Oxford University Press. ISBN: 0195176782.

32. **Chung, P.W.H., Cheung, L., Stader, J., Jarvis, P., Moore, J. and Macintosh, A., (2003),** "Knowledge-Based Process Management - an Approach to Handling Adaptive Workflow", *Journal of Knowledge-Based Systems*, **16,** pp.149–160.

33. **Chung, S. H. and Snyder, C. A., (2000),** "ERP Adoption: a Technological Evolution Approach", *International Journal of Agile Management Systems*, **2** (1), pp. 24-32.

34. **Chung, S. H. and Snyder, C. A., (1999),** "ERP Initiation: A Historical Perspective", *Proceedings of the 5th Americas Conference on Information System*, Milwaukee, WI, USA, 13-15.08.99, pp.213-215.

35. **CIMdata, (2002),** "Product Lifecycle Management - Empowering the Future of Business", by CIMdata, Inc
http://www.cimdata/publications/
(Site visited July 2003)

36. **CIMdata, (2001),** "Collaborative Product Definition Management (cPDm): An Overview", A CIMdata Report, available to download at: http://www.cimdata/publications/cPDm_Def.pdf
(Site visited December 2006)

37. **CIMdata, (1997),** "Product Data Management", Report for the UK Department of Trade and Industry, http://www.dti.gov.uk/ (search for CIMdata).

38. **Ciocoiu, M., Gruninger, M. and Nau, D.S., (2001),** "Ontologies for Integrating Engineering Applications", *Journal of Computing and Information Science in Engineering*, **1**(1), pp.2-22.

39. **Clarke, I., Hong, T.W., Miller, S.G., Sandberg, O. and Wiley, B., (2002),** "Protecting Free Expression Online with Freenet", *IEEE Internet Computing,* **6**(1), pp.40-49.

40. **Clarke, I., Sandberg, O., Wiley, B. and Hong, T.W., (2000),** "Freenet: A Distributed Anonymous Information Storage and Retrieval System" in Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability, LNCS 2009, edited by Hannes Federrath. Springer: Berlin.

41. **Cloutier, L., Frayret, JM., D'Amours, S., Espinasse, B. and Montreuil, B., (2001),** "A Commitment-Oriented Framework for Networked Manufacturing Co-ordination", *International Journal of Computer Integrated Manufacturing,* **14**(6), pp.522 – 534.

42. **CORBA, (2006),** http://www.corba.org/ (Site Visited December 2006).

43. **Corcho, O., Fernandez-Lopez, M. and Gomez-Perez, A., (2003),** "Methodologies, Tools and Languages for Building Ontologies. Where is their meeting point?", *Data & Knowledge Engineering,* **46**, pp.41–64.

44. **Davenport, T.H., (1993),** "Process Innovation: Reengineering Work through Information Technology", Harvard Business School Press.

45. **Davenport, T.H. and Prusak, L., (1998),** "Working Knowledge: How Organisations Manage What They Know", Boston, MA: Harvard Business School Press, p.5.

46. **Davies, J., Fensel, D. and van Harmelen, F., (2002),** On-To-Knowledge: Semantic Web enabled Knowledge Management, John Wiley and Sons Ltd, ISBN: 0470848677.

47. **Denny, M., (2004a),** "Ontology Editor Survey 2004", http://www.xml.com/2004/07/14/examples/Ontology_Editor_Survey_2004_Table_-_Michael_Denny.pdf.
(Site visited December 2006)

48. **Denny, M., (2004b),** "Ontology Tools Survey, Revisited", http://www.xml.com/lpt/a/1447
(Site visited December 2006)

49. **Denscombe, M., (1998),** The Good Research Guide for Small-Scale Social Research Projects, Scarborough, Great Britain: Biddles Ltd, ISBN 0335198066.

50. **Dorador, J. M. and Young, R. I. M., (2000),** "Application of IDEF0, IDEF3 and UML Methodologies in the Creation of Information Models", *International Journal of Computer Integrated Manufacturing*, **13**(5), pp.430-445.

51. **Duineveld, A.J., Stoter, R., Weiden, M.R., Kenepa, B. and Benjamins, V.R., (2000)**, "WonderTools? - A Comparative Study of Ontological Engineering Tools", *International Journal of Human-Computer Studies*, **52,** pp.1111-1133.

52. **Dvir, R and Evans, S., (1998)**, "Mapping Knowledge Reuse in R&D Processes - The Implementation of Quality Tools", *Proceedings of the 12th International Conference of the Israel Society for Quality*, Heidelberg, Berlin: Springer.

53. **Earl, M.J., (1990),** "Putting IT in its Place: A polemic for the Nineties", Working Paper, Oxford Institute of Information Management Research and Discussion Papers: pp.100-108.

54. **Ehrig, M. Tempich, C. and Staab, S., (2003)**, "SWAP: Ontology-based Knowledge Management with Peer-to-Peer Technology", *4th European Workshop on Image Analysis for Multimedia Interactive Services WIAMIS*, Queen Mary University, London.

55. **Esteves, J. and Pastor, J., (2001)**, "Enterprise Resource Planning Systems Research: An Annotated Bibliography", *Communications of the Association for Information Systems*, 7(8), pp.1-52.

56. **Ettighoffer, D., (1992)**, L'Entreprise Virtuelle ou Les Nouveaux Modes de Travail, (The Virtual Enterprise on the New Working Methods), Paris: Odiles Jacob, ISBN: 2-7381-0163-1.

57. **Eyler, A.N., (2001)**, Corporate Knowledge Management, ProTopics. http://www.deloitte.com/dt/cda/doc/content/JulyAugustEnglish.pdf (Site visited August 2004)

58. **Fiorano Software, (2003)**, "A Whitepaper in Super-Peer Architectures for Distributed Computing", Fiorano Software, Inc. http://whitepapers.silicon.com/0,39024759,60087713p-39000463q,00.htm. (Site visited May 2004)

59. **Garetti, M., Terzi, S., Bertacci, N. and Brianza, M., (2005)**, "Organisational Change and Knowledge Management in PLM implementation", *International Journal of Product Lifecycle Management*, **1**(1), pp.43-51.

60. **Goasdoue, F., (1999)**, "A Knowledge Based Approach for Information Integration: The PICSEL System", Dagstuhl-Seminar-Report 251, Page 7.

61. **Goddard, W., (1985)**, "MRP II makes it work", *Canadian Controls and Instrumentation*, **24**(3), pp.44-56.

62. **Gold-Bernstein, B. and Ruh, W., (2004),** Enterprise Integration: The Essential Guide to Integration Solutions, Published by Addison-Wesley, ISBN: 032122390X.

63. **Gong, L., (2002),** "Project JXTA: A Technology Overview", Published by Sun Microsystems, Inc.

64. **Goossenaerts, J., and Pels H.J., (1998),** "Product related Data and Knowledge Management in the Intelligent Enterprise', *Proceedings of IMS-Europe*, Lausanne.

65. **Gourlay, S., (2000),** "Frameworks for Knowledge: a Contribution towards Conceptual Clarity for Knowledge Management", *The Proceedings of Knowledge Management: concepts and controversies conference*, (Warwick, UK: University of Warwick.

66. **Gunasekaran, A., (1998),** "Agile Manufacturing: Enablers and an Implementation Framework", *International Journal of Production Research*, **36**(5), pp.1223 – 1247.

67. **Hall, G., (2000),** "Engineering Information Management in the 21st Century", *Information Management & Technology*, **33**(5), pp.212–216.

68. **Hausser, R., (2000),** "The Four Basic Ontologies of Semantic Interpretation", *The Tenth European - Japanese Conference on Information Modelling and Knowledge Bases*, Finland, pp.21-40.

69. **Hlupic, V., Pouloudi, A. and Rzevski, G., (2002),** "Towards an Integrated Approach to Knowledge Management: 'hard', 'soft', and 'abstract' Issues", *Knowledge and Process Management*, **9**(2), pp.90-102.

70. **Holland, C. P. and Light, B., (1999),** "A Critical Success Factors Model for ERP Implementation", IEEE Software, **16**(3), pp.30-36.

71. **Holzner, S., (2001),** Inside XML, New Riders Publishing, ISBN: 0735710201, Indianapolis, Indiana.

72. **Hou, C.S.J., Noy, N.F. and Musen, M.A., (2002),** "A Template-Based Approach Toward Acquisition of Logical Sentences", *Intelligent Information Processing 2002, World Computer Congress*, Montreal, Canada, pp.77-89.

73. **Huang, G.Q., (2002),** "Web-based Support for Collaborative Product Design Review", *Computers in Industry*, **48**(1), pp.71–88.

74. **Huang, G.Q, Huang, J. and Mak, K.L., (2000)**, "Agent-based Workflow Management in Collaborative Product Development on the Internet", *Journal of Computer-Aided Design*, **32**, pp.133-144.

75. **Huang, G.Q., Yee, W.Y. and Mak, K.L., (2001),** "Development of a Web-based System for Engineering Change Management", *International Journal of Robotics and Computer Integrated Manufacture*, **17**(3), pp. 255–267.

76. **IBROW3, (2004),**
http://www.swi.psy.uva.nl/projects/IBROW3/home-ibrow.html.
(Site Visited October 2004)

77. **ITAG, (1999)**, "The Knowledge Economy", A Submission to the New Zealand Government by the Minister for Information Technology's IT Advisory Group, August, ISBN 0-478-23435-X.
http://www.med.govt.nz/pbt/infotech/knowledge_economy/
(Site Visited October 2004)

78. **Jarvinen, P., (1991)**, "On Approaches in Information Systems Research", *in Proceedings of the 14th Information Systems Research In Scandinavia*, Umea, Sweden, 11-14.08.91, pp. 1-13.

79. **Java Software, (2006)**, "Object Serialization", http://java.sun.com/j2se/1.4.2/docs/guide/serialization/ (Site Visited December 2006)

80. **Jenz & Partner, (2003)**, "Document of Frequently ask Questions in Ontology", Jenz & Partner GmbH. http://www.jenzundpartner.de/FAQ_Ontology.pdf (Site Visited May 2005)

81. **Kessler, J., (1991)**, "MRP II: in the Midst of a Continuing Evolution - Industrial Engineering", **23**(3), pp. 38-40.

82. **Krepchin, I. P., (1986)**, "Make MRP II Work with Real-Time Links - Modern Materials Handling", **41**(10), pp.111-114.

83. **Laguda, A., (2002)**, "Aggregate Assembly Process Planning for Concurrent Engineering", PhD Thesis, University of Durham, School of Engineering, Durham, UK.

84. **Laird, C., (2001)**, "XMI and UML Combine to Drive Product Development", Vice president, Phaseit Inc. http://www-106.ibm.com/developerworks/xml/library/x-xmi/ (Site Visited May 2005)

85. **Leuf, B., (2003)**, Peer to Peer: Collaboration and Sharing over the Internet, Addison Wesley Professional, ISBN: 0201767325, Pages: 464; Edition: 1st.

86. **Li, H., Li, X.Y. and Gu, J., (2006)**, "A new algorithm of similarity measuring for multi-experts' qualitative knowledge based on outranking relations in case-based reasoning methodology", *Intelligent Data Engineering and Automated Learning - 7th International Conference on Intelligent Data Engineering and Automated Learning* 2006, pp.637-644, ISBN: 3-540-45485-3.

87. **Li, W.D., Ong, S.K., Wong, Y.S., Fuh, J.Y.H., Lu, Y.Q. and Nee, A.Y.C., (2004a),** "Feature-based Design in a Collaborative and Distributed Environment", *Computer-Aided Design*, **36**(9), pp.775–797.

88. **Li, W.D., Wong, Y.S. and Fuh, J.Y.H., (2004b),** "An Internet-enabled Integrated System for Co-design and Concurrent Engineering", *Computers in Industry*, **55**(1), pp.87–103.

89. **Lin, H.K. and Harding, J.A., (2003)**, "An Ontology Driven Manufacturing System Engineering Moderator for Global Virtual Enterprise Teams", *Proceedings of the 1st International Conference on Manufacturing Research* (ICMR), Glasgow, UK, pp.365-370.

90. **Linthicum, D.S., (2003),** Next Generation Application Integration: From Simple Information to Web Services, Published by Addison Wesley, ISBN: 0201844567

91. **Little, D. and Lee, H. L., (1999),** "Survey of PDM systems in the UK SMEs", *Proceedings of the 15th IPR*, Limerick, Ireland, 9–13 August.

92. **Liu, D.R. and Shen, M., (2003)**. "Business-to-business Workflow Interoperation based on Process-Views", 2003, *Journal of Decision Support Systems*, **8**(3), pp.23-45.

93. **Liu, D.T. and Xu X.W., (2001),** "A Review of Web-based Product Data Management systems", *Computers in Industry*, **44**, pp.251– 262.

94. **Luscombe, M., (1993),** "MRPII: Integrating the Business: A Practical Guide for Managers", Oxford: Butterworth-Heinemann Ltd.

95. **Madhusudan, T., (2005),** "An Agent-based approach for Coordinating Product Design Workflows", *Computers in Industry*, **56**(3), pp.235–259.

96. **Mahdjoubi, D. and Harmon, G., (2001),** "Knowledge Management: a Conceptual Platform for the Sharing of Ideas", *Proceedings of the 64th ASIST Annual Meeting,* Washington, DC 2001. pp.290-304, Medford, NJ: Information Today.

97. **Malamateniou, F. and Vassilacopoulos, G., (2003)**, "Developing a Virtual Patient Record using XML and Web-based Workflow Technologies", *International Journal of Medical Informatics*, **70**, pp.131-139.

98. **Maropoulos, P.G. and Gao, J.X., (2000),** "Case for Support: Distributed and Collaborative Product Development and Manufacturing Knowledge Management (EPSRC GR/R26757)", Durham and Cranfield, November 2000.

99. **Maropoulos, P. G., Yao, Z. and Bradley, H. D., (1998),** "CAPABLE: an Aggregate Process Planning System for Integrated Product Development", *Journal of Materials Processing Technology,* **76** (1-3), pp.16-22.

100. **Maropoulos, P. G., Yao, Z., Bradley, H. D. and Paramor, K. Y. G., (2000a),** "An Integrated Design and Planning Environment for Welding Part 1: Product Modelling", Journal of Materials Processing Technology, **107** (1-3) pp.3-8.

101. **Maropoulos, P. G., Yao, Z., Bradley H. D. and Paramor, K. Y. G., (2000b),** "An Integrated Design and Planning Environment for Welding Part 2: Process Planning", *Journal of Materials Processing Technology*, **107** (1-3), pp.9-14.

102. **Maropoulos, P.G., Bramall, D.G. and McKay, K.R., (2003a),** "Assessing the Manufacturability of Early Product Designs using Aggregate Process Models", *Proceedings of the Institution of Mechanical Engineers: Part B: Journal of Engineering Manufacture*, **217** (9), pp.1203-1214.

103. **Maropoulos, P.G., Bramall, D.G., McKay, K.R., Rogers, B.C. and Chapman, P., (2003b),** "An Aggregate Resource Model for the Provision of Dynamic Resource-aware Planning", *Proceedings of the Institution of Mechanical Engineers: Part B: Journal of Engineering Manufacture,* **217** (10), pp.1471-1480.

104. **Martensson, M., (2000),** "A Critical Review of Knowledge Management as a Management Tool", *Journal of Knowledge Management,* **4**, pp. 204-216.

105. **McIntosh, K.G., (1995),** Engineering Data Management: A Guide to Successful Implementation, McGraw-Hill Book Company, ISBN: 0-07-707621-4 / 0077076214.

106. **McKay, K. R., Bramall, D. G., Colquhoun, P. G. and Maropoulos, P. G., (2001a),** "Product, Process and Resource Manufacturing Models for a Distributed Process Planning System", *Proceedings of the 5th IASTED Conference in Applied Simulation and Modelling.*

107. **McKay, K. R., Bramall, D. G., Colquhoun, P. G. and Maropoulos, P. G., (2000),** "Manufacturing Analysis of Conceptual and Embodiment Aerospace Designs: An Aggregate Product Model Specification", *16th International Conference on Computer Aided Production Engineering (CAPE 2000),* Edinburgh.

108. **McKay, K.R., Bramall, D.G., Rogers, B.C., Chapman, P., Cheung, W.M., and Maropoulos P.G., (2003),** "Design Change Impact Analysis during Early Design Specification," *Int. Journal of Computer Integrated Manufacturing,* **16**(7-8), pp.598–604.

109. **McKay, K. R., Colquhoun, P. G., Bramall, D. G. and Maropoulos, P. G., (2001b),** "An Aggregate Resource Model for the Provision of Digital Mock-up within a Distributed Manufacturing Planning System", *Proceedings of the 2nd International Conference on Advances in Production Engineering,* Warsaw, Poland, pp.131-140.

110. **McLaughlin, B. and Loukides, M., (2001),** Java and XML (O'Reilly Java Tools), O'Reilly, 1st Edition, ISBN: 0596000162.

111. **Miller, E., (2003),** "State of the PLM Industry", *CIMData PLM Conference 2003*, MI, USA, 23rd July.

112. **Moench, E., (2003)**, "Semantic Miner: Ein integratives Ontologie-basiertes Knowledge Retrieval System", *Proceedings of the 2nd Conference on Professional Knowledge Management – Experiences and Visions (WM2003)*, Switzerland, pp.25-32.

113. **Moench, E., Ullrich, M., Schnurr, H.P. and Angele, J., (2003),** "SemanticMiner- Ontology-Based Knowledge Retrieval", *Journal of Universal Computer Science*, **9** (7), pp.682-696.

114. **Moore, C. and Graham, G., (2002),** Streamline Your Business Processes With Workflow & Extranet Solutions, 2001, extracted from - The Workflow Handbook 2002, published in association with the Workflow Management Coalition (WfMC), ISBN 0-9703509-2-9.

115. **Moore, J., Stader, J., Chung, P., Jarvis, P. and Macintosh, A., (1999),** "Ontologies to Support the Management of New Product Development in the Chemical Process Industries", *12th International Conference on Engineering Design (ICED 99)*, Munich, pp.159-164.

116. **Nonaka, I. and Takeuchi, H., (1995),** The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation, New York: Oxford University Press, (Pages.58)

117. **OTK, (2004),** http://www.ontoknowledge.org/index.shtml, (Site Visited May 2005).

118. **Oluic-Vukovic, V., (2001),** "From information to knowledge: some reflections on the origin of the current shifting towards knowledge processing and further perspective", *Journal of the American Society for Information Science and Technology*, 52, pp. 54-61.

119. **OMG (Object Management Group), (2005),** http://www.omg.org/technology/documents/formal/xmi.htm (Site Visited March 2005).

120. **OMG (Object Management Group), (2003),** Unified Modelling Language http://www.omg.org (Site Visited December 2003).

121. **Ontoprise, (2004),** http://www.ontoprise.de/products/ontobroker_en (Site Visited March 2004).

122. **OntoBroker, (2004),** http://www.ontoprise.de/products/ontobroker_en (Site Visited March 2004).

123. **Oppenheim, A.N., (1992),** Questionnaire Design, Interviewing and Attitude Measurement, Publisher: Pinter Pub Ltd, New Education Edition (July 1992), ISBN: 1855670437.

124. **Paashuis, V., (1998),** The Organisation of Integrated Product Development, Hardcover: Pages 277, Publisher: Springer-Verlag (Nov 1997), ISBN: 3540762256.

125. **Pawar, K. and Sharifi, S., (2000),** "Virtual collection of Design Teams: Coordinating for speed", *International Journal of Agile Management Systems*, **2**(2) pp.104-113.

126. **Penserini, L., Liu, L., Mylopoulos, J., Panti, M. and Spalazzi, L., (2003),** "Cooperation Strategies for Agent-Based P2P Systems", International Journal of Web Intelligence and Agent System, **1** (1) pp. 3-21.

127. **Philpotts, M., (1996),** "An Introduction to the Concepts, Benefits and Terminology of Product Data Management", *Industrial Management & Data Systems,* MCB University Press (ISSN 0263-5577), CIM data Ltd 1996, **96**(4), pp. 11–17.

128. **Plesums, C., (2002)**, Introduction to Workflow, extracted from - The Workflow Handbook 2002, published in association with the Workflow Management Coalition (WfMC), March 2002, ISBN 0-9703509-2-9.

129. **Plossl, G. W., (1995)**, Orlicky's Material Requirements Planning. 2nd Edition, New York: McGraw-Hill, 1995.

130. **Portella, J., (2000),** "Collaborative Management of the Product Definition Lifecycle for the 21st Century", CD-Rom Proceedings, *PDT Europe Conference*, Noordwijk, Netherlands, May 2000.

131. **Porter, M. E., (1985),** Competitive Advantage: Creating and Sustaining Superior Performance, Free Press, London, pp. 557.

132. **Protégé2000, (2003),** "Knowledge-Based System Development Tools", http://protege. stanford.edu/
(Site Visited January 2004).

133. **PTC Windchill, (2002),** User Guide 6.0, Parametric Technology Corporation, 2001–2002, Release 6.0, T857-60-01.

134. **Puchalski, E.R., (1997),** "ERP helps Manufacturers Control their Businesses", *I&CS,* **70**(6), pp.63-66.

135. **Qiang, L., Zhang, Y.F. and Nee, A.Y.C., (2001),** "A Distributed and Collaborative Concurrent Product Design System through the WWW/Internet", *International Journal of Advanced of Manufacturing Technology,* **17**(5), pp.315–322.

136. **RDF Vocabulary Description Language 1.0, (2004),** http://www.w3.org/TR/2004/REC-rdf-schema-20040210/ (Site Visited December 2006)

137. **Rich, E. and Duchessi, P., (2001),** "Models for Understanding the Dynamics of Organizational Knowledge in Consulting Firms", *Proceedings of the 34th Hawaii International Conference on System Sciences,* Maui, Hawaii, (HICSS-34) **3**, pp.3007-3014.

138. **Richardson, J. R., (1998),** "MRP II Paves the Road to Computerized Planning systems", *CIM Review,* **4**(2), pp.15-19.

139. **Risson, J. and Moors, T., (2006),** "Survey of research towards robust peer-to-peer networks: Search methods", *Computer Networks,* **50**(17), pp.3485-3521.

140. **Roche, C., (2000),** "Corporate Ontologies and Concurrent Engineering", *Journal of Materials Processing Technology,* 107, pp.187-193.

141. **Rodriguez, K. and Al-Ashaab, A., (2005),** "Knowledge Web-based System Architecture for Collaborative Product Development", *Computers in Industry,* **56**, pp.125-140.

142. **Sackett, P.J. and Bryan, M.G., (1998),** "Framework for the Development of a PDM Strategy", *International Journal of Production and Operations Management,* **18**(2), pp.168-179.

143. **Salustri, F., (2006),** "Opensource Product Lifecycle Management Navigator Project", Available on line at: http://deseng.ryerson.ca/xiki/View/Oplm/RugPlotPaper (Site visited, 05 May 2006).

144. **Satine Consortium, (2003),** "SATINE1: Semantic-based Interoperability Infrastructure for Integrating Web Service Platforms to Peer-to-Peer Networks for Travel Industry", European Commission through IST-1-002104-STP SATINE, January 2003.

145. **Saya, I. M., Casat, F., Dayal, U. and Shan, M.C., (2001),** "Integrating Workflow Management Systems with Business-to-Business Interaction Standards", Software Technology Laboratory, HP Laboratories Palo Alto, HPL-2001-167, 2001.

146. **Schlenoff, C., Gruninger, M., Tissot, F., Valois, J., Lubell, J. and Lee, J., (2000),** "The Process Specification Language (PSL) Overview and Version 1.0 Specification," *National Institute of Standards and Technology*, NISTIR 6459, Gaithersburg, MD.

147. **Schmidt, W., (2000),** "Knowledge Management and Decision Support", Proceedings of The Knowledge Workplace: Transforming How and Where We Work, 13-15 March 2000, San Antonio, TX.

148. **Schmuller, J., (1999),** SAMS Teach Yourself UML in 24 Hours, Sams Publishing, ISBN 0-672-31636-6.

149. **SCRA, (2003),** "A White paper in the STEP Manufacturing Suite version 3", http://isg-scra.org/STEP/files/STEP_MfgSuiteWhitePaper.pdf (Site Visited December 2006).

150. **SemanticMiner., (2004),** http://www.ontoprise.de/products/semanticminer_en. (Site Visited February 2004).

151. **Shah, J. J. and Mantyla, M., (1995),** Parametric and feature-based CAD/CAM: concepts, techniques, and applications. New York, USA: Wiley. Pages 619, ISBN 0-471-00214-3.

152. **Siple, M., (1998),** The Complete Guide to Java Database Programming: JDBC, ODBC & SQL, (McGraw-Hill, Maidenhead), ISBN: 0-079-13286-3.

153. **Smith, P.G. and Reinertsen, D. G. (1995),** Developing Products in half the Time New York: van Nostrand Reinhold, ISBN: 0442020643.

154. **Sowa, J.F., (1995),** "Top-level Ontological Categories", *International Journal of Human Computer Studies*, **43**, pp.669-685.

155. **Srikanntaiah, T.K. and Koenig, M.E.D., (2000),** Knowledge Management for the Information Professional. Medford, NJ: Information Today.

156. **Stark, J., (2004),** Product Lifecycle Management: 21st century Paradigm for Product Realisation, Publisher: Springer; 1st Edition, ISBN: 1852338105.

157. **Stevens, P., (2003),** "Small-Scale XMI Programming: A Revolution in UML Tool Use?" *Journal of Automated Software Engineering,* **10**(1), pp.7-21.

158. **Streatfield, D. and Wilson, T.D., (1999),** "Deconstructing knowledge management", *Aslib Proceedings*, **51**(3), pp.67-72.

159. **Su, X. and Ilebrekke, L., (2002),** "A Comparative Study of Ontology Languages and Tools", Proceedings of Advanced Information Systems Engineering: *14th International Conference, CAiSE 2002,* Toronto, Canada, May 27-31, Page 761.

160. **Tachbrook, (1999),** http://www.tachbrook-consulting.com/ (Site Visited October 2002)

161. **Tunnicliffe, C.N. and Maropoulos, P.G., (2003),** "Re-engineering the Part Cost Management Process for an Automotive Manufacturer by Implementing a Real Time Web-based Data Warehouse", *Proceedings of the 1st International Conference on Manufacturing Research (ICMR),* Glasgow, UK, pp.345-350.

162. **Turban, E. and Aronson, J.E., (2000)**, Decision Support Systems and Intelligent Systems, US Imports & PHIPEs, ISBN: 0130327239.

163. **Valarakos, A., Paliouras, G., Karkaletsis, V. and Vouros, G., (2004)**, "Enhancing Ontological Knowledge through Ontology Population and Enrichment", *Proceedings of the 14th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2004)*, Springer Verlag, 3257, pp.144-156.

164. **Van Heijst G., Schreiber A.T. and Wielinga B.J., (1997)**, "Using Explicit Ontologies in Knowledge-based Systems Development", *International Journal of Human-Computer Studies (IJHCS)*, **46**, pp. 183–291.

165. **Vernadat F.B., (2002)**, "Enterprise Modelling and Integration (EMI): Current status and Research Perspectives", *Annual Reviews in Control, Publisher: Elsevier Science*, **26** (1), pp. 15-25.

166. **W3C, (1999)**, "Resource Description Framework (RDF) Model and Syntax Specification".
http://www.w3.org/TR/1999/REC-rdf-syntax-19990222
(Site Visited March 2002).

167. **Whittaker, B., (1999)**, "What went wrong? Unsuccessful Information Technology Projects", Information Management & Computer Security, **7**(1), pp.23–29.

168. **Wikipedia, (2006)**
http://en.wikipedia.org/wiki/Tacit_knowledge
(Site Visited December 2006)

169. **Wilson, T.D., (2002)**, "The Nonsense of Knowledge Management", *International Journal in Information Research*, **8**(1), ISSN 1368-1613.

170. **Xiao, A., Choi, H.J., Kulkani, R., Allen, K.J., Rosen, D. and Mistree, F., (2001),** "A Web-based Distributed Product Realisation Environment", *Proceedings of ASME 2001 Design Engineering Technical Conferences*, Pittsburgh, PA, USA, DETC00/CIE-14624.

171. **Xie, Y.L. and Salvendy, G., (2003),** "Agent-based features for CAD browsers in foster engineering collaboration over the Internet", *International Journal of Production Research,* **41**(16), pp.3809–3829.

172. **Xu, X.W. and Liu, T., (2003),** "A web-enabled PDM system in a Collaborative Design Environment", *International Journal of Robotics and Computer Integrated Manufacturing,* **19**(4), pp.315–328.

173. **Yan, J., Yang, Y. and Raikundalia, G. K., (2003)**, "A Data Storage Mechanism for Peer-to-Peer Based Decentralised Workflow Systems", *Proceedings of 15th International Conference on Software Engineering and Knowledge Engineering,* San Francisco, USA, pp.354-358.

174. **Yao, Z., Bradley, H.D. and Maropoulos, P.G., (1998)**, "An Aggregate Weld Product Model for The Early Design Stages", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* **12,** pp.447-461.

175. **Yendluri, P., (2000),** "RosettaNet Implementation Framework (RNIF) 2.0", webMethods, Inc.

176. **Zhao, J., Cheung, W.M., Young, R.I.M. and Bell, R., (1999),** "An Object Oriented Manufacturing Data Model for a Global Enterprise", *15th International Conference on Computer-Aided Production Engineering (CAPE'99),* Durham, UK, 19-21st April, 1999, pp.582-588, ISBN 0-9535558-0-1.

APPENDIX A – PRODUCT LIFECYCLE MANAGEMENT

Appendix A shows a comprehensive description of the core areas and their capabilities of the PLM system.

Core Areas	***Capabilities***	***Description***
Product Data Management (PDM)	*Product Structure*	Bill-of-material (BOM)
	CAD Integration	-
	Engineering Change	A function uses to manage changes and modifications of engineering design associate with manufacturing, equipment, material.
	Specification Management for Process	Supports creation, manage, exchange and standardize of product specifications for manufacturing processes and makes it available for business partners in the supply chain.

Core Areas	***Capabilities***	***Description***
Product Data Management (PDM)	*Recipe Management*	Recipes control the manufacturing equipment. Each atomised production step is controlled by one or several recipes. In a complex production a variety of recipes have to be administrated. Recipes change often. Besides different variations, different versions develop by and by. Often the result is an unmanageable surge of different recipes which have to be administrated and provided in the manufacturing process. This requires a central recipe management that is able to provide automatically the proper recipe at the exact position anytime.
Product Data Management (PDM)	*Configuration Management:-*	To enable manufacturers of complex, heavily regulated or highly customizable products to ensure that the product delivered meets the customer's exact specifications. It also ensures that uniqueness between configurations is always tracked and identified.

Core Areas	*Capabilities*	*Description*
		A strategic function to manage a complex product which consists of hardware, electronics and software. Unlike the traditional PDM system which used for Mechanical CAD only.
Collaborative Product Development (CPD)	*System Engineering*	The solution delivers competitive advantage by optimizing the early stages of product engineering, when companies establish the requirements, functions, and technical architecture of a new product, and commit 80 percent of its cost. System engineering also enables companies to integrate product design with product definition, ensuring 'right to market' delivery through traceability between final product design and initial customer requirements. Finally, it enables maximal reuse of valuable corporate knowledge to help cut development time and costs.

Core Areas	***Capabilities***	***Description***
Collaborative Product Development (CPD)	*Product Design Tool*	Design collaboration, share design, analysis design (i.e. drag and drop of different components.)
	Process Design Tool	-
	Visualization	-
	CAD Interoperability	-
	Formulation	Collaborative design process is often about sharing formulation data, i.e. this behaves like the application of 'knowledge management system based on text data than the 3-d geometry environment of mechanical CAD.
Direct Material Sourcing (DMS)	*RFG Workflow*	Request for quote selling, buying and purchasing.
	Quote Analytics	As volume of quoting increases so as the number of suppliers i.e. strategic analysis to find the most appropriate vendors for outsourcing.
	Purchasing Integration	-
	Part Database	-

Core Areas	***Capabilities***	***Description***
Product Portfolio Management (PPM)	*Portfolio Analytics*	You need to analyze your product portfolios and to assess the feasibility of new product concepts.
	Program Management	Used to manage product data through out different stages of its lifecycle
	Resource Management	identify sources and obtain resources needed to support product development activities; coordinate the supply, allocation, distribution, and delivery of resources so that they arrive where and when most needed; and maintain accountability for the resources used
	Stage-Gate-Process	The stage gate process was designed to manage the flow of projects into the development process i.e. automated program updates based on product lifecycles.

Core Areas	***Capabilities***	***Description***
Product Portfolio Management (PPM)	*Executive Dashboard*	A function which allows the *company executive* to look at a product, and to know, right away, whether a product destined to boost or break the company's bottom line from its earliest stages of development, or even before, when it is simply being contemplated. In this fantasy, a green light signals *projected customer acceptance*, and *on budget* and *on schedule* products, and the executive is armed with critical information that enables the company to devote its limited, financial and human capital resources to only those products most likely to succeed.

APPENDIX B – PRODUCT DATA MANAGEMENT

Appendix B shows a comprehensive description of the core features and individual functionalities of the PDM system.

Product Data Management		
Features	***Functions***	***Description***
User Functions	*Data Vault and Document Management*	Data Vault and Document Management provides secure storage and retrieval of product definition information. Check-in and Check-out functions work with data stored in an electronic vault to provide secure storage and access control. Release levels for design data are defined and users are assigned access authorizations. Meta-data stores information about product data so that changes, release levels, approval authorizations, and other data controls can be tracked and audited. The meta-data also if used to create relationships among product data so that information can be grouped and related by common usage and among products. Non-electronic data can be managed by reference although they cannot have the same level of security as for data in the vault. The electronic vault either contains the product information itself or information that allows users to access that data, but the user need not know where data are stored.

Features	*Functions*	*Description*
User Functions	*Workflow and Process Management*	Workflow and Process Management provide a route to drive a business with information. With Workflow and Process Management, a PDM system can interact with people, working according to predefined business processes of an organization and with data and documents, to achieve corporate objective. Workflow and process define and control changes o product data. The workflow and processes are defined in terms of a sequence of events that must occur before modified product data are allowed to be release. PDM tracks approvals and authorizations of changes. The change process is a sequence of controlled events. Changes are accepted by electronic sign-off. Appropriate information is routed automatically. Audit and historical records are maintained.

Features	*Functions*	*Description*
User Functions	*Workflow and Process Management*	Workflow and Process Management provide a route to drive a business with information. With Workflow and Process Management, a PDM system can interact with people, working according to predefined business processes of an organization and with data and documents, to achieve corporate objective. Workflow and process define and control changes o product data. The workflow and processes are defined in terms of a sequence of events that must occur before modified product data are allowed to be release. PDM tracks approvals and authorizations of changes. The change process is a sequence of controlled events. Changes are accepted by electronic sign-off. Appropriate information is routed automatically. Audit and historical records are maintained.

Features	*Functions*	*Description*
User Functions	*Products Structure Management*	Product Structure Management facilitates the creation and management of product configurations and Bill of Material (BOM). PDM systems allow users and applications to link or associate product definition data such as drawings, documents, and process plans to parts and product structures. Product Structure Management can track physical components and their connections, or virtual objects such as a cooling system or an emergency escape procedure. PDM systems allow views that can show structural relationships, manufacturing processes; documentation, financial, support and repair, and other relationships embodied in product definition data. BOMs can be transferred to PDM systems and also MRP managed information can be transferred to PDM systems.

Features	*Functions*	*Description*
User Functions	*Classification*	Classification of parts allows similar or standard parts, processes, and other design information to be grouped by common attributes and retrieved for use in products. This leads to greater product standardization, reduced re-design, saving in purchasing and fabrication, and reduced inventories. Classification and retrieval of parts (and other kinds of objects) can be facilitated by the development and management of classification structures. For parts, these structures can be part family hierarchies. To make good use of part family management capabilities, an organization must also have a definition of the part family hierarchy that best fits its own industry and operations. Standard libraries of parts such as fasteners, electronic components, pipe components, or building materials are available and can be accessed via similar mechanisms.

Features	***Functions***	***Description***
User Functions	*Program Management*	Program Management provides work breakdown structures (WBS) and allows resource scheduling and project tracking. Resources and managed data are linked to provide an added level of planning and tracking. A key advantage stems from the ability to relate the WBS tasks to the PDM system's knowledge of approval cycles and product configurations. Tasks required to complete the product program are ordered within the work breakdown structure so resources and the project schedule can be monitored. As the project progresses from task to task, expended resources are recorded against the plan. Completion of the data required from each task is reported through the approval process.

Features	***Functions***	***Description***
Utility Functions	*Communication and Notification*	On-line, automated notification of critical events means that all personnel are informed concerning the current state of the project. Electronic mail is used to notify people about important events or required actions on-line. Delays and misplaced communications are minimized. A software mechanism as a "trigger" is used to spawn notifications and other actions automatically. Almost any event in the product development process can cause a message to be sent or cause another event to occur. Triggers are used to streamline data vault and document management and workflow and process management procedures.

Features	*Functions*	*Description*
Utility Functions	*Data Transport*	All data are stored and accessed under control of the PDM system, so a user need not know where in the computer network data are stored the system keeps track of data locations and allows users to access it knowing only a data set's name. Names of data sets are not limited by the local computer's file-naming conventions so they can be more meaningful. In fact, the user need not know how to use the computer's file and directory commands. Moving data from one location to another or from one application to another is an operation that the PDM system performs-users don't need to be concerned with operating system and network commands.

Features	*Functions*	*Description*
Utility Functions	*Data Translation*	The system administrator can pre-define data translators to be used to convert data between pairs of applications and to formats for various display and output devices. This allows enforcement of standard data forms. While translators may not be provided as an integral part of the PDM system, they may be applied because the system knows the data format of each controlled file and which translation is appropriate. Triggers can cause data to be translated automatically from one application to another at appropriate times. Thus, the correct data are more likely in any situation.
Utility Functions	*Image Services*	Raster, vector, and video images are treated the same as any other data by the PDM system. On-line access is provided to a wide range of previously difficult-to-distribute product information providing this information in a structured manner to more users. Images Services accelerate processes such as Engineering Change Requests (ECR) and Orders (ECO) by facilitating the exchange of information and comments among users.

Features	Functions	Description
Utility Functions	*System Administration*	The administrator sets up the operational parameters of the PDM system and monitors its performance. Most systems can be tailored to conform to corporate standards and to improve the efficiency of operations for individual users. In addition to customary user interface customization, the operational features of PDM systems can be tailored. Many PDM systems provide standard, off-the-shelf interfaces to popular applications such as CAD, CAM, MRP/ERP, technical publishing, and office automation. In addition, most systems offer interface toolkits, GUI builders, and application interface toolkits.
Web-based Collaboration	*Parts and Supplier Information*	Web-based architecture used across product lines, business units, and the supply chain to consolidate and standardizes part and supplier information in order to reduce global procurement and product development costs hence, making it easy to identify reusable and commercially available parts and preferred suppliers.

Features	*Functions*	*Description*
Web-based Collaboration	*Product and Process Information*	By using a Web-based infrastructure with a federated architecture to create a strategic source of product and process information and presents it in a collaborative environment where all members of the value chain can communicate.
Web-based Collaboration	*Inter-Enterprise*	By using of Web access that facilitates customer, supplier, and partner collaboration to deliver innovative products through the Internet.
e-Integration	*Collaborative Product Commerce*	This allow customers, partners, and suppliers to collaborate in a Web environment to create innovative new products, deliver those products to market faster, and manage the complexities of an evolving supply chain. It does this through its unique web-centric approach and federated architecture.
e-Integration	*Interactive Selling Environment*	A customer-facing B2B solution for suppliers of parts and components delivered through the Web that changes the tools of selling to provide a more dynamic, interactive selling environment

Features	*Functions*	*Description*
Electronic Vault or Data repository	*Legacy documents*	Such as drawings on paper or aperture cards, or other hardcopy records can be captured by scanning and storing them within the electronic vault as images. Some users prefer to convert images of textual documents by optical character recognition (OCR) into computer-readable text. It is feasible to convert some raster (scanned) images of drawings into vector data via software, while some users prefer to capture vector data by digitization techniques.
Electronic Vault or Data repository	*An electronic vault*	Used as a repository to control all kinds of product information. The vault of a data store that contains some data within itself and controls other externally generated data by managing access to it. Two types of data are stored: 1. Product data generated in various applications, such as specifications, CAD models, CAE data, maintenance records, and operating manuals 2. Meta-data, which is data about PDM-controlled information. Meta-data is stored in a PDM database and supports the functions performed by the PDM system

APPENDIX C – PDM PRODUCT ASSESSMENT

Appendix C1 shows the PDM providers and their offerings. The survey was carried out at June 2002. These were the most popular PDM systems during the survey. C2 describes the result of the evaluation of the commercial PDM systems.

C1. PDM PRODUCT PROVIDERS

		Vendors	IBM & Dassault	PTC	Unigraphic (EDS PLM)	Agile Software	CoCreate	Eigner & Partners	SDRC (EDS PLM)	SmarTeam	Auto-trol	Enabled Systems (UK)	MatrixOne
		Products	ENOVIApm	Windchill	iMAN	Agile Workplace	WorkManager	Axalant 2000	Metaphase	SmarTeam	Centra 2000	ESysPDM	eMatrix
		Vendor Offering	16	16	16	15	15	15	15	15	13	13	13
Functions													
User Functions	Data Vault and Document Management		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Workflow and Process Management (Engineering Change)		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Product Structure Management		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Part Classification and Group Technology		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Project and Program Structure Management		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Utility Functions	Communication Notification		yes		yes	yes	yes	yes	yes	yes	yes		yes
	Data Transport		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Data Translation		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Engineered-to-Order		yes										
	Design-to-Order			yes									
	Contract Manufacturing Product Data			yes	yes								
	Image Services		yes	yes	yes	yes	yes	yes	yes	yes		yes	
	System Administration		yes	yes	yes	yes	yes	yes	yes	yes		yes	
User Environment	User Interfaces (Team based software development etc)		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Application Interfaces (CAD,NC Part Programs, ERP etc)		yes	yes	yes	yes	yes	yes	yes	yes	yes		yes
Electronic Vault (Data repository)	Data from other Computer Applications		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	Legacy Documents		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
	An Electronic Vault		yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes

		Vendors	Eigner & Partners	CoCreate	PTC	SmarTeam	Unigraphic (EDS PLM)	IBM & Dassault	MatrixOne	SDRC (EDS PLM)	Agile Software	Auto-trol	Enabled Systems (UK)
		Products	Axalant 2000	WorkManager	Windchill	SmarTeam	iMAN	ENOVIApm	eMatrix	Metaphase	Agile Workplace	Centra 2000	ESysPDM
		Vendor Offering	5	4	4	4	4	3	3	2	1	1	0
Web-Based Collaborative	parts and supplier information		yes	yes	yes	yes	yes	yes	yes	yes			
	product and process information		yes	yes	yes	yes	yes	yes	yes	yes	yes		
	Inter-Enterprise Collaboration		yes	yes	yes	yes	yes	yes	yes			yes	
e-Integration	B2B Solution for Collaborative Product Commerce (CPC)		yes	yes		yes							
	B2B Solution for Interactive Selling Environment		yes		yes		yes						

		Vendors	Eigner & Partners	MatrixOne	SDRC (EDS PLM)	SmarTeam	PTC	Auto-trol	Enabled Systems (UK)	Unigraphic (EDS PLM)	Agile Software	CoCreate	IBM & Dassault
		Products	Axalant 2000	eMatrix	Metaphase	SmarTeam	Windchill	Centra 2000	ESysPDM	iMAN	Agile Workplace	WorkManager	ENOVIApm
		Vendor Offering	3	3	3	3	2	2	2	2	1	1	1
Modularity and Customization	Multi-language Support					yes							
	Tailoring Tools and Customization		yes	yes	yes	yes	yes	yes	yes	yes			yes
Operating Environment	Client-Server					yes			yes				
	Open-Network Support		yes	yes	yes		yes						
	Open, Flexible Architecture		yes	yes								yes	
	Open Standards-based Architecture				yes			yes		yes	yes		

Companies	Product Name	Product References
IBM & Dassault	ENOVIApm	http://www-3.ibm.com/solutions/plm/pub1/05256965005a58c0/5/06b30bfeeae4e1f587256a400056707a.jsp
PTC	Windchill	http://www.ptc.com/uk/index.htm
Unigraphics (EDS PLM)	iMAN	http://www.plmsolutions-eds.com/products/iman/
Agile Software	Agile Workplace	http://www.agilesoft.com/corporate/contact.asp
CoCreate	Work Manager	http://www2.cocreate.com/cocreate/index.cfm?CategoryNodeID=D913F3AA-A673-11D4-97DE00508BEEE4DB&submit=1
Eigner & Partners	Axalant 2000	http://edms-service.web.cern.ch/edms-service/axalant.html
SDRC (EDS PLM)	Metaphase	http://www.sdrc.com/metaphase/index.shtml
SmarTeam	SmarTeam	http://www.3ds.com/products-solutions/plm-solutions/enovia-smarteam/overview/
Auto-trol	Centra 2000	http://www.centra2000.com/Product_Information/default.htm
Enabled Systems	ESysPDM	http://www.enabled-systems.com/default.asp?pageid=2
MatrixOne	eMatrix	http://www.matrix-one.com/products/ematrix.html

C2. EVALUATION OF THE FUNCTIONALITY OF COMMERCIAL PDM SYSTEMS

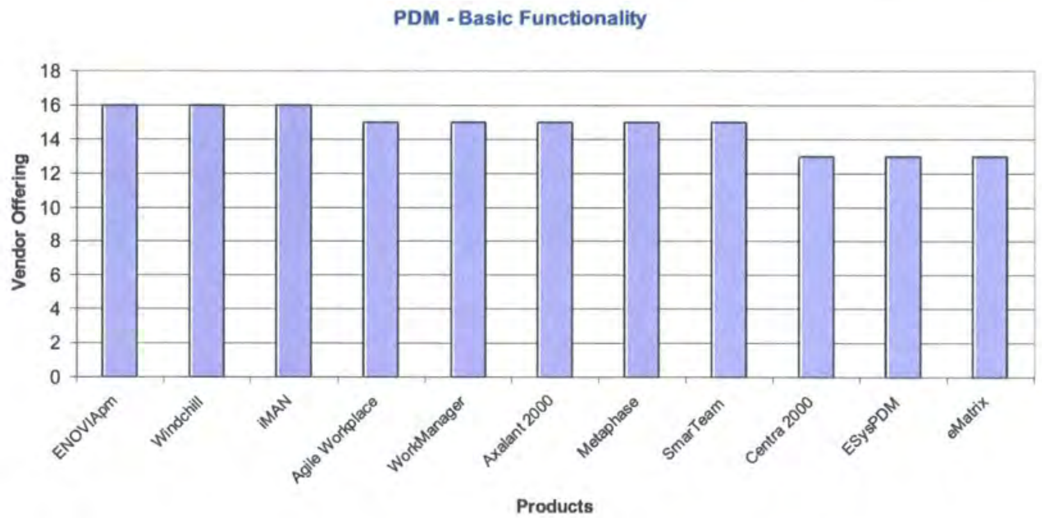
Tables C3, C4 and C5 illustrate the functionalities offered by various PDM products. The diagrams indicate the number of offerings provided by various products. These functionalities are categorised as follows:

- Basic functionality (User, Utility and Electronic Vault)
- Web-based Collaborative and e-Integration
- Customization and Operating Environment Functionalities

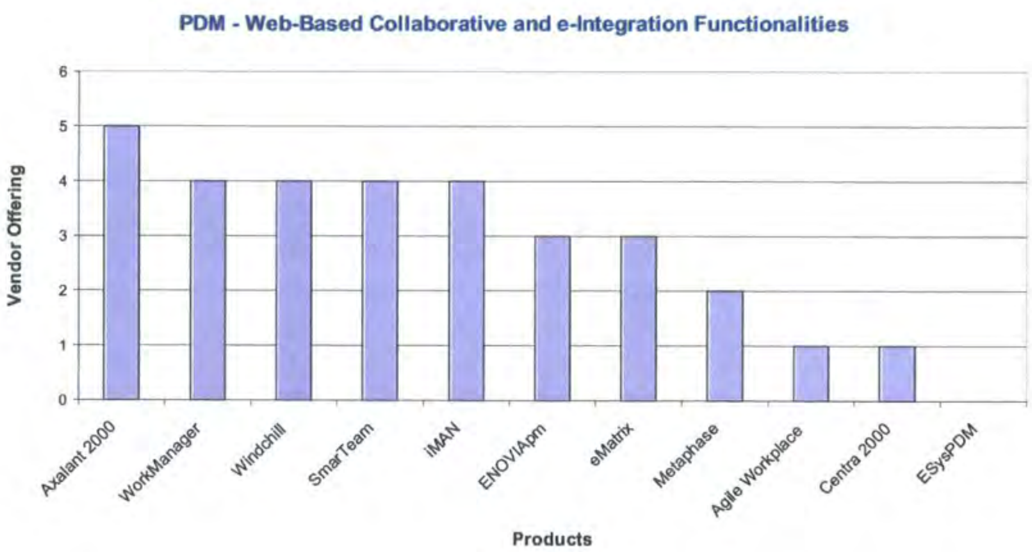
The figures give an overall comparison of each technology category of all product providers. The vertical axis shows the number of functions offers by individual software providers which are also used to determine the ranking of the software providers in terms of how mature is their system.

The assessments also provide some useful information to establish the pros and cons in terms of selecting the right system for particular needs. In general, many systems offer more complex solutions on top of the basic "Workflow Management" and "Project Management" functionality. Additional functionalities include the capability to integrate into different types of CAD and ERP as well as offering Web-based Collaborative Environment solutions.

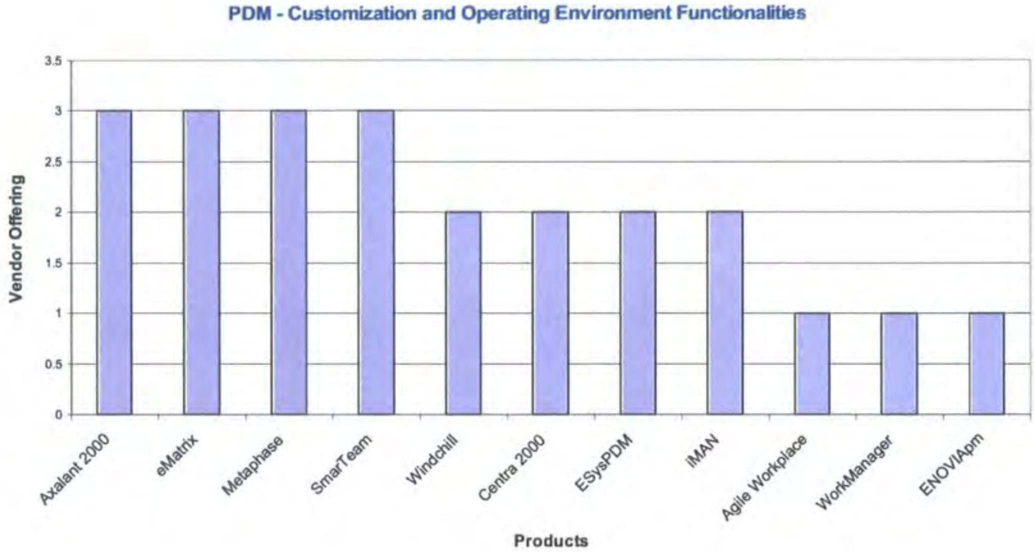
C3: PDM BASIC FUNCTIONALITY



C4: PDM WED-BASED AND E-INTEGRATION FUNCTIONALITIES



C5: PDM – CUSTOMIZATION AND OPERATING ENVIRONMENT FUNCTIONALITIES



APPENDIX D - COMPANIES LISTING FOR PDM INDUSTRIAL SURVEY

Appendix D shows a list of the respondent companies in the industrial survey.
NE - NorthEast of England
SY and WY - South and West of Yorkshire

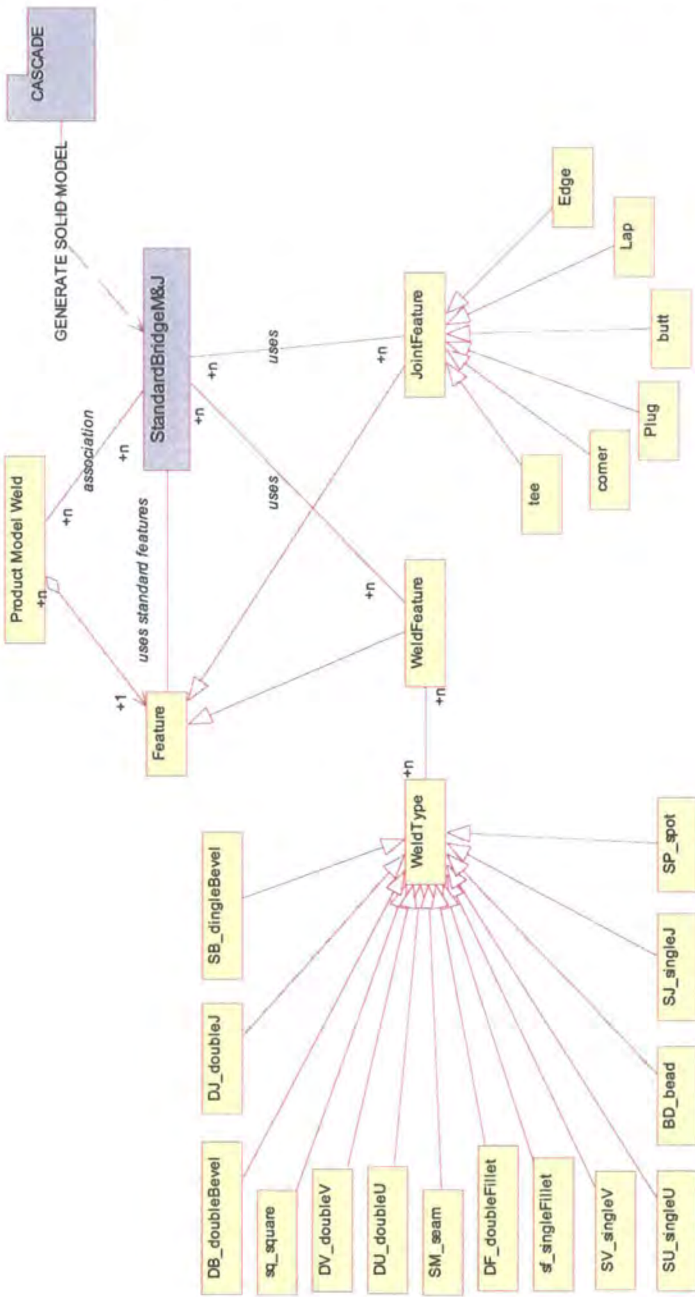
	Region	Name	Company Size	Industry	PDM Systems	More Information
1.	NE	Aalco	Large	Engineering Raw Material	eMatrix	www.aalco.co.uk/index.html
2.	NE	AEI CABLES LIMITED	Large	Cable Manufactures for the MOD Industry	Smarteam	www.aeicables.co.uk
3.	WY	Armstrong	Large	Manufacturer of Floor and Ceiling	Smarteam	www.armstrong.com/
4.	NE	BAE SYSTEMS LAND SYSTEMS	Large	MOD and Defence	eMatrix	www.baesystems.com/
5.	NE	Black & Decker	Large	Power Tools	Workmanager	www.blackanddecker.co.uk
6.	NE	CAV Aerospace Ltd	Large	Aero Structures	UGS Teamcentre	www.cav-aerospace.net/
7.	WY	Du Pont (UK) Ltd	Large	Paint & Coatings	ENOVIApm and Workmanager	www.dupont.com
8.	SY	Euchner (UK)	Large	Automation and Safety Switches	Axalant EDM	www.euchner.com/
9.	WY	Huddersfield Valves Ltd	Large	Specialising in valves, actuators and controls.	iMAN	www.huddersfieldvalves.co.uk
10.	WY	Lowe Engineering	Large	Vessel design and Fabrication	Smarteam	www.loweengineering.com/
11.	SY	MEC	Large	Electrical Distributed Systems	iMAN	www.mecbirmingham.com
12.	NE	NEL Technologies	Large	Components and Sub–assemblies	ENOVIApm	www.nel-ltd.co.uk/pages/welcome.php
13.	NE	Nissan Motor Manufacturing (UK) Ltd	Large	Automotive	ENOVIApm	www.nissan.co.uk/

	Region	Name	Company Size	Industry	PDM Systems	More Information
14.	NE	Plexus	Large	Electronic system for defence	eMatrix (as client to BAE)	www.plexus.com
15.	NE	Siemens Power Generation	Large	Compressors, industrial Processes	Centra 2000	www.siemens.co.uk/index.jsp
16.	SY	AEC	Medium	Manufacturer and design of automotive electronics	EDM	www.aeceuro.co.uk/
17.	SY	AGEMASPARK LIMITED	Medium	Automation Machines	NA	www.agemaspark.co.uk
18.	NE	Durham Precision Engineering	Medium	Precision Machined Components	Smarteam	www.durham-precision-engineering.co.uk
19.	NE	Phoenix Precision Electric Ltd	Medium	Defence for BAE system	Smarteam	www.phoenix-elec.com
20.	NE	Scott-Matrix	Medium	Polymer Products	Teamcentre	www.scott-matrix.com/
21.	NE	Vickers Pressing	Medium	Pressings - Sheet Metal, Component Assembly	EDM	www.bodyinwhitepressings.co.uk/
22.	NE	Victor	Medium	Mining equipment	Smarteam	www.victor.co.uk/mining/index.htm
23.	NE	Advance Automated Systems Ltd	Small	UK conveyor manufacturer	EDM	www.advanceautomation.co.uk/
24.	WY	AMT	Small	Machining	ESysPDM	www.shotblastuk.com/
25.	NE	NIM Engineering Ltd	Small	Hydraulic Equipment	ESysPDM	unknow
26.	NE	Tungsten Alloys Ltd	Small	Alloy Manufacturer, supplying blanks	eMatrix (as client to BAE)	www.tungsten-alloys.co.uk

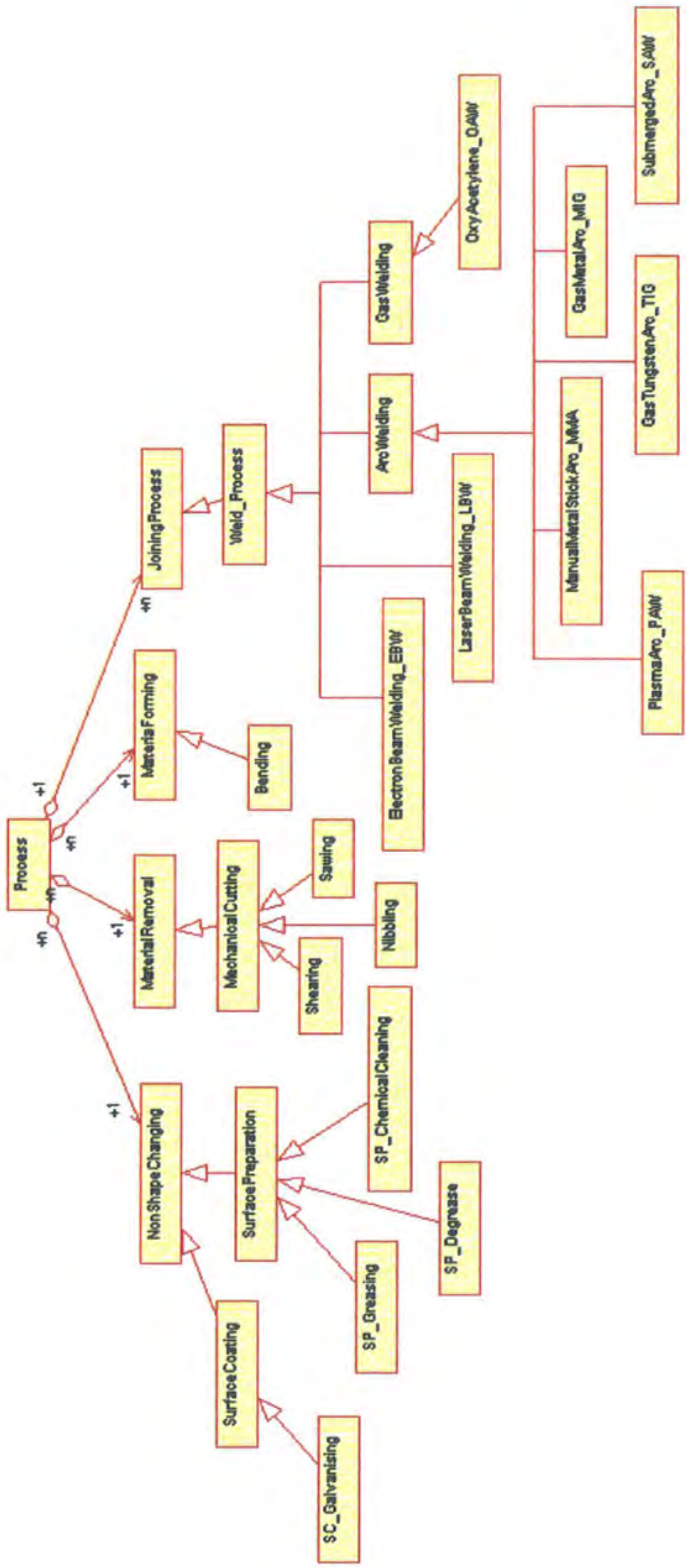
APPENDIX E – CLASS DIAGRAMS FOR INUDTRIAL COLLABORATORS PRODUCT AND MANUFACTURABILITY

Appendix E illustrates the UML Class diagrams of the additional capabilities for the selected exemplars used in case studies 1 and 2 in product, process and factory resource models.

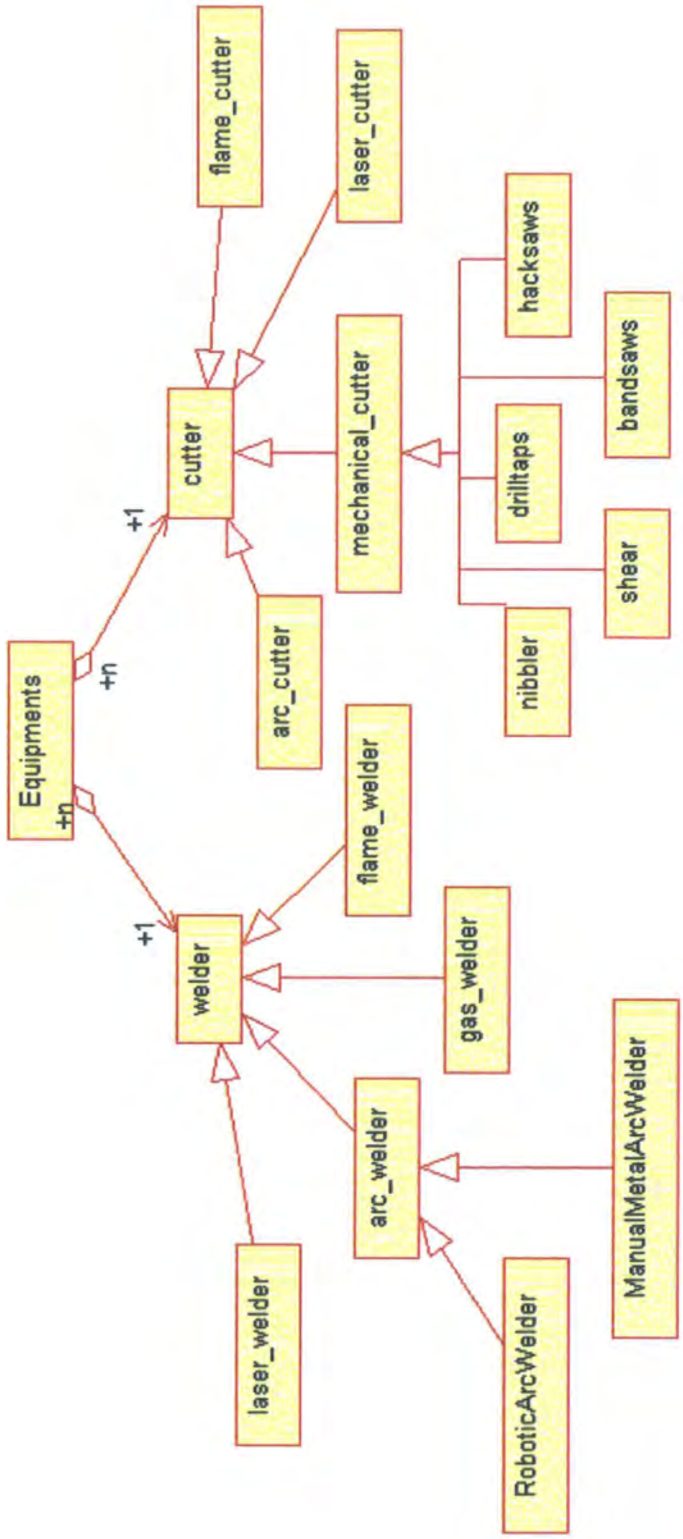
E.1 MABEY & JOHNSON PRODUCT MODEL ON WELDING



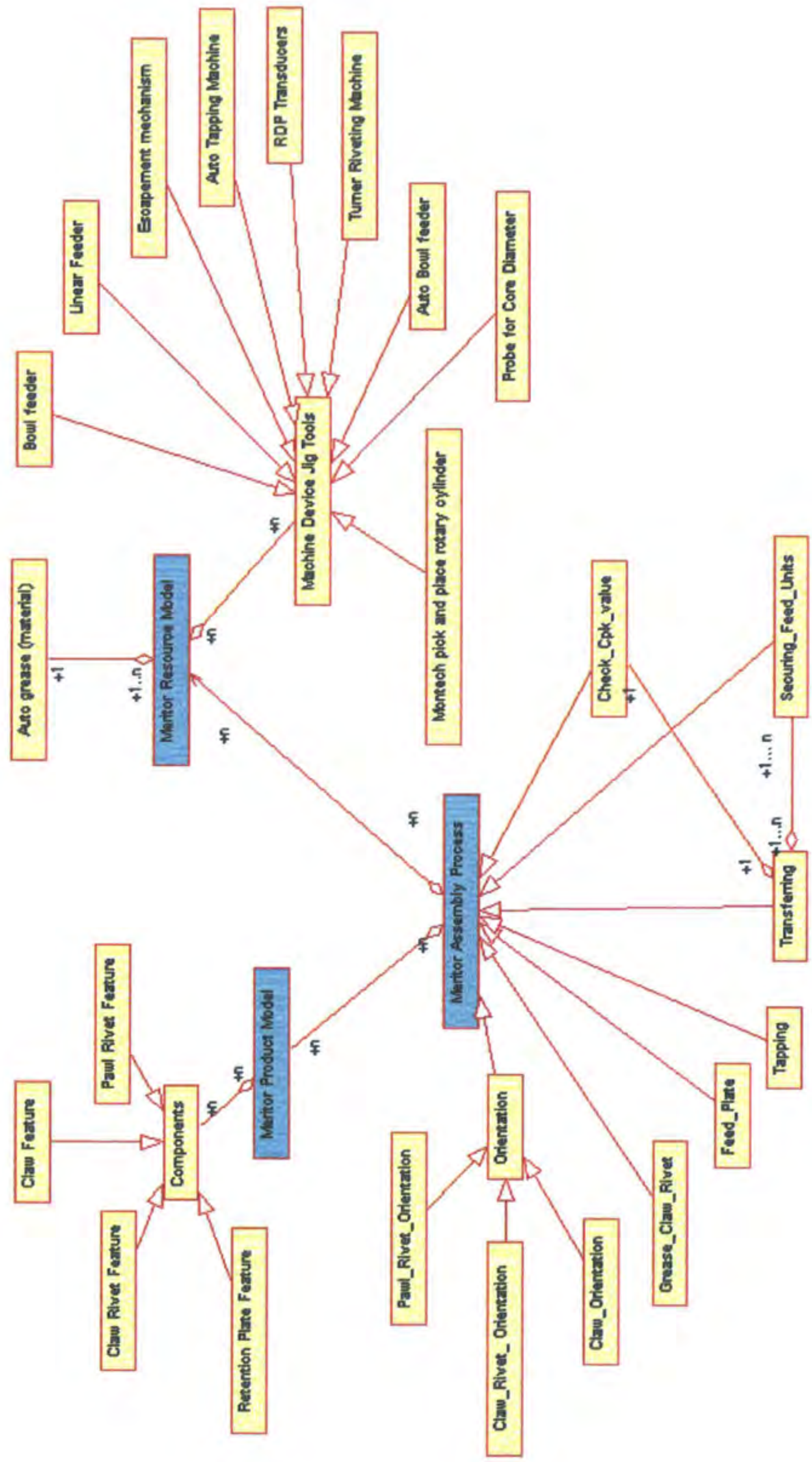
E.2 MABEY & JOHNSON PROCESS MODEL ON WELDING



E.3 MABEY & JOHNSON RESOURCE MODEL ON WELDING



E.4 ARVINMERITOR PRODUCT, PROCESS AND RESOURCE MODELS BASED ON FOUR COMPONENTS



APPENDIX F – MABEY & JOHNSON MANUFACTURING KNOWLEDGE

Appendix F illustrates the data collected for the Organisational knowledge-based system.

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Design**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Written and Benchmarking	12/08/03	Design	Steve Moore	Design features with access and drainage involves galvanising process	*0.75*	Design features which aid the access and drainage of molten zinc will improve the quality and appearance of the coating:- The benchmarking of good design requires for filling, venting and drainage requires:- • Means for the access and drainage of molten zinc • Means for escape of gases from internal compartments (venting)
	12/08/03	Design	Steve Moore	Design of holes and venting involves galvanising process	*0.75*	Design of holes for venting and draining should be as large as possible. The minimum hole diameters are given in the following table:- Size of hollow section (mm) — Minimum Diameter of hole (mm) <25 — 10 ≥25-50 — 12 >50-100 — 16 >100-150 — 20 >150 — Consult galvanizer
	12/08/03	Design	Steve Moore	Design important point	*0.75*	Designer bear in mind that plain carbon steel, some low-alloy steels and iron and steel castings can be galvanized.
	12/08/03	Design	Steve Moore	Design important point	*0.75*	Design tolerance, the working conditions of the relationship of parts are vital (Boothroyd Dewhurst Design for Manufacturing and Assembly).

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Design**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Observation and Intuition	12/08/03	Design	Steve Moore	Design important point	*0.75*	Designer should bear in mind that soldered or brazed components should not be galvanized.
	12/08/03	Design	Steve Moore	Detail Design	*0.75*	Detailed design advice is available from Galvanizers Association.
	12/08/03	Design	Steve Moore	Design corners	*0.75*	Internal and external stiffeners, baffles, diaphragms, gussets etc, the designer should take into account that design of corners should cropped to aid the flow of molten zinc.
Employee Tacit and Experience	12/08/03	Design	Steve Moore	Design important point	*0.75*	Considerations to design the steel bridge, issues such as temperature range for expansion allowances. Loading, span-of-bridge and stress factors for design characteristics in structure safety and yield
	12/08/03	Design	David Halliday	Design important point	*0.75*	It is important bear in mind that the steelwork is immersed into a bath of molten zinc at a temperature 450 deg.C. Thus, any feature which aid the access and drainage of molten zinc will improve the quality of the coating and reduce costs.
	12/08/03	Design	David Halliday	Design important point	*0.75*	For complete protection, molten zinc must be able to flow freely to all surfaces of a fabrication.

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Design**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Best Practice	2/08/03	Design	Steve Moore	Design of tube	*0.75*	*Explosion may occur* if air is trapped in the design, for example a tube.
	12/08/03	Design	Steve Moore	weldments	*0.75*	weldments (weld joints) should be designed to *avoid acid traps*
	12/08/03	Design	Steve Moore	Design Flat panel	*0.75*	Flat panels should be braced to minimise *the risk of distortion*
	12/08/03	Design	David Halliday	Design involves in fabrication	*0.75*	A fabrication consisting a variety of materials with different surface conditions should be avoided as this could affect uniformity and appearance of coating in galvanising process.
	12/08/03	Design	David Halliday	Design fasteners	*0.75*	*Design of fasteners* to accommodate the thickness of zinc when galvanizing threaded components, extra clearance must be provided on the female threads.
	12/08/03	Design	David Halliday	Design overlap components	*0.75*	In designing components which requires galvanising, overlap surfaces should be avoided as far as possible. i.e. if overlaps are completely sealed by welding there is a risk of explosion during dipping due to increased pressure of any entrapped air.
	12/08/03	Design	David Halliday	Design and residual stress	*0.75*	Efforts can be made at the design stage to minimise residual stresses:- ▪ Controlling welding procedures during fabrication ▪ Arranging weld seams symmetrically. The size of weld seams should be kept to minimum. ▪ Avoiding large changes in structural cross-section which may increase distortion and thermal stresses in galvanizing

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Manufacturing Processes**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Written and Benchmarking	12/03/04	Product Development	Keith Newman	All Processes	*0.75*	Important points to ensure that: ▪ senior managers support benchmarking and are committed to continuous improvements ▪ the objectives are clearly defined ▪ sufficient resources are available to complete projects within a certain time scale ▪ - benchmarking teams have a clear picture of their organisation's performance before approaching others for comparisons
	12/03/04	Product Development	Keith Newman	All Processes	*0.75*	It is important to AVOID ▪ benchmarking for the sake of it ▪ expecting that benchmarking will be quick or easy ▪ When approaching the process for the first time, it is worthwhile learning from others who have built up experience of applying benchmarking in a comparable way ▪ Focusing entirely on comparisons of performance measures rather than the processes and activities that enable the achievement of good practice.

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Manufacturing Processes**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor** *(default value)*	**Knowledge Statements**
Written and Benchmarking	12/03/04	Product Development	Keith Newman	All Processes	*0.75*	Benchmarking Process Guide below: 1) Planning 2) Collecting data and information 3) Analysing the findings 4) Recommendations
	12/03/04	Galvanising	Brian Stahler	Galvanising	*0.75*	Powder Coating is method of adding colour to metal surfaces. Like galvanizing it is carried out under carefully controlled conditions in a factory. For this reason the maximum size steel fabrication to be powder coated will be limited. (see BS 6497:1981(1991) or European standard (EN 13438)
Observation and Intuition	12/03/04	Galvanising	Brian Stahler	Galvanising	*0.75*	The galvanizing process is actually quite quick and virtually anything can be galvanized within 24 hours. If anyone tells you different don't accept it.
	12/08/03	Galvanising	Brian Stahler	Galvanising	*0.75*	If welding slag is not removed in the cleaning process and may result in black bare spots after hot dip galvanising.
	12/08/03	Galvanising	Brian Stahler	Galvanising	*0.75*	Water soluble paint or detachable metal labels can be used for temporary identification marks on fabrication.

Mabey and Johnson Ltd, Liley Industrial Estate, UK				**Manufacturing Processes**		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Employee Tacit and Experience	12/08/03	Robotic and Manual Welding	Andy Chiswell	Galvanising	*0.75*	make sure apply Surface Cleaning on Weld joints
	12/08/03	Galvanising	Brian Stahler	Galvanising	*0.75*	Factors which influence the thickness and appearance of the galvanized coating include: ▪ Chemical composition of the steel ▪ Bath immersion time ▪ Bath withdrawal rate ▪ Steel surface condition ▪ Cold working of steel prior to galvanizing ▪ Steel cooling rate
	12/08/03	Galvanising	Brian Stahler	Galvanising	*0.75*	If certain areas of steel work need to remain uncoated this can be achieved by masking, using high temperature tape, grease or paint.
	12/08/03	Galvanising	Brian Stahler	Galvanising	*0.75*	Clean steel surfaces are an essential requirements for good hot dip galvanising

Mabey and Johnson Ltd, Liley Industrial Estate, UK				Manufacturing Processes		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Best Practice	12/08/03	Product Development	Keith Newman	5 S	*0.75*	The 5S are housing keeping, workplace organization, cleanup, keep cleanliness and discipline
	12/08/03	Product Development	Keith Newman	Demand Flow	*0.75*	It is a concept to pull raw materials and products through the process strictly according to the dictates of customer demand.
	12/08/03	Product Development	Keith Newman	JIT	*0.75*	Just-in-time is a management philosophy that strives to eliminate sources of manufacturing waste by producing the right part in the part place at the right time
	12/08/03	Product Development	Keith Newman	Lean Business	*0.75*	Lean Business is a set of business practices that eliminates non-value adding. lean business ensures that the value chain, the set of steps that actually add value to the desired product is strongly aligned.
	12/08/03	Product Development	Keith Newman	Six Sigma	*0.75*	Originated from Motorola for total-quality-management practice on 'zero-defects'.

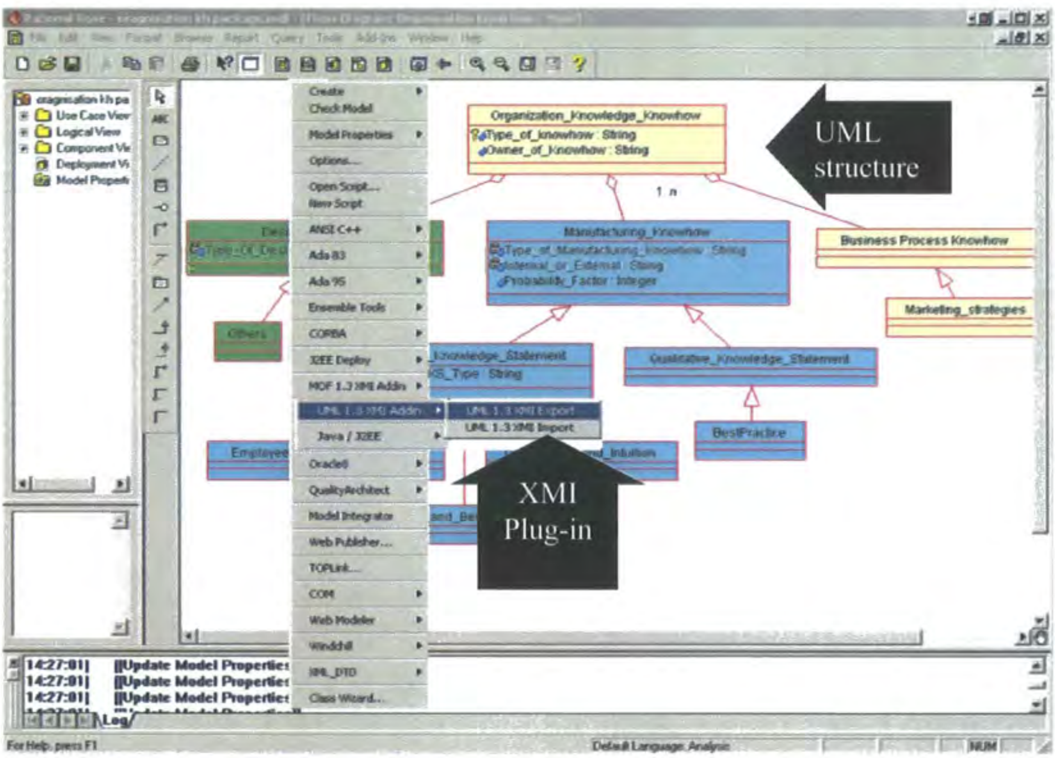
Mabey and Johnson Ltd, Liley Industrial Estate, UK				Factory Resources		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Written and Benchmarking	12/03/04	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	Make sure web segment is clean
	12/03/04	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	ABB Robots with National Standard E70S-6, 045 1.2 mm weld wire. Also the 045 weld wire size selected is not appropriate for parts < 2 mm.
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	The bottom line; when welding parts < 2 mm its imperative the plant uses a 035 or 040 wire.
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	Part dimension and TCP control is critical when welding gage applications > 2mm. Typically on parts > 2 mm we would deal with weld fusion issues, misplaced welds, weld profiles and start stops concerns
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	The pulsed "arc length sensitivity" is a serious concern as it makes the pulsed mode unsuitable for most robot welds in which high weld speeds > 100 mm/min is a key criteria.

Mabey and Johnson Ltd, Liley Industrial Estate, UK				Factory Resources		
Knowledge Statement Type	**Date Prepared**	**Group**	**Owner of Know How**	**KS Type**	**Probability Factor *(default value)***	**Knowledge Statements**
Observation and Intuition	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	*Butt / Fillet gage welds*, *Lap welds* do allow higher weld current as the two thickness combine to absorb the weld heat.
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	With robot single pass "butt" welds on parts 1.2 to 1.6 mm, we typically would require short circuit transfer welds made at 150 to 200 amps. For these thin applications it's logical to use an 035 - 040 wire.
Employee Tacit and Experience	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	For any robot weld application that uses a weld current over 200 amps, rather than use the pulsed mode, advice is the use of the high deposition, stable spray transfer mode.
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	With gage parts 1.7 - 1.8 mm, the *butt weld current* would be increased to approx. 200 to 240 amps, again note that *lap welds* will allow higher weld current but it's logical to keep the weld current close to the butt max current to avoid weld burn through. *Butt welds* on 2 mm parts could be robot welded using 240 - 280 amps.
	12/08/03	Robotic and Manual Welding	Andy Chiswell	Robotic Welding important	*0.75*	An 045 MIG wire and argon CO2 mixes requires a minimum of 250 - 260 amps to attain stable spray transfer, most 3/16 and ¼ fillet welds are welded between 260 and 350 amps.

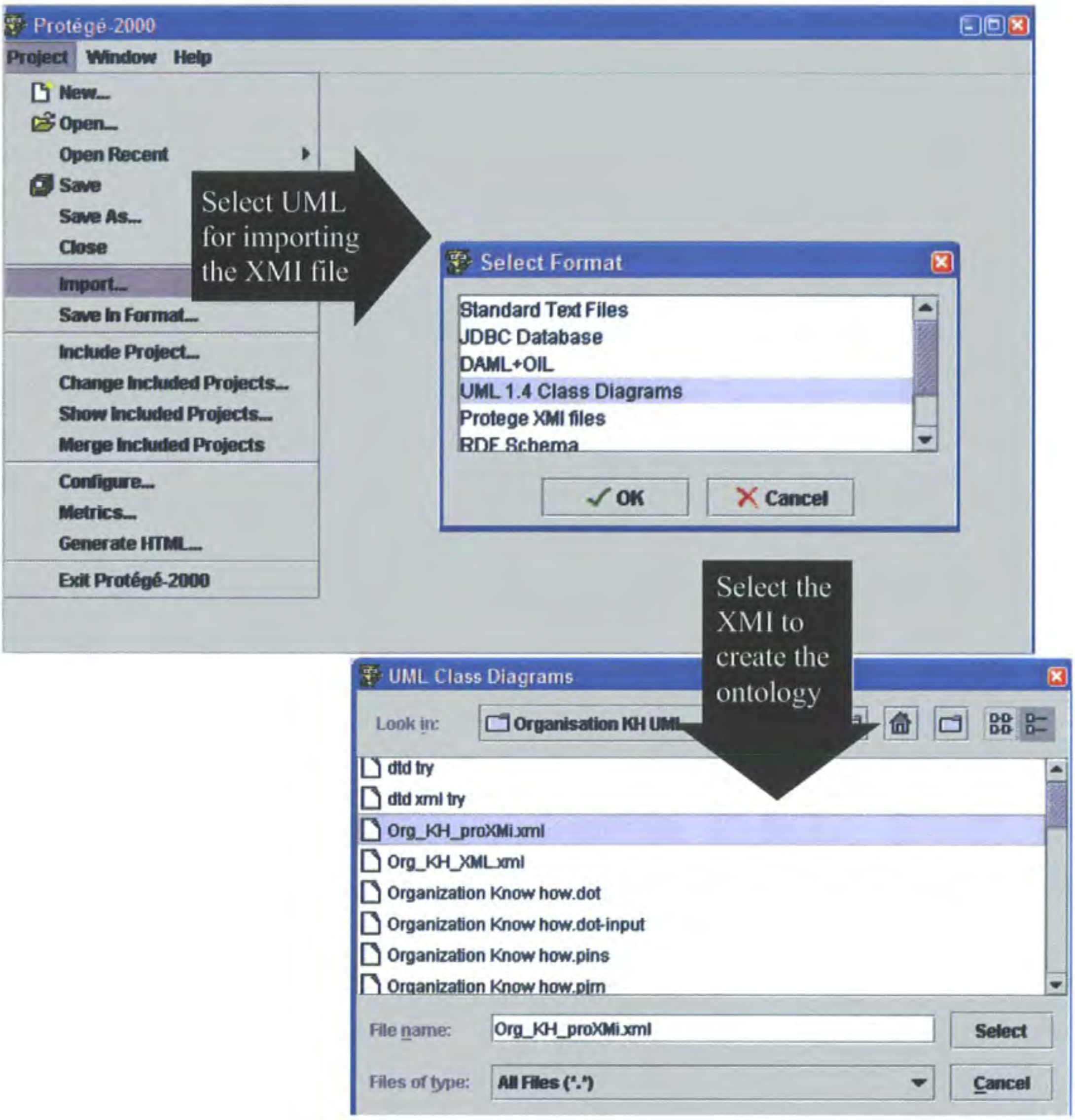
APPENDIX G – USING XMI TO CREATE THE ONTOLOGY

Appendix G shows the application of XML Meta data Interchange (XMI) to create an ontology. Figure (a) shows the steps of exporting a UML structure using the XMI plug-in. Figure (c) shows the XMI file of the ontology and (b) illustrates the steps of importing the XMI file into the protégé2000 knowledge editor to configure and refine the ontology.

G1. EXPORT THE UML CLASS DIAGRAM USING XMI



G2. IMPORT XMI INTO THE PROTÉGÉ2000 KBS EDITOR



G3. THE ORGANISATIONAL KNOWLEDGE ONTOLOGY IN XMI FORMAT


```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <XMI xmi.version="1.2" timestamp="Tue Aug 05 09:17:01 BST 2003">
  - <XMI.header>
    - <XMI.documentation>
        <XMI.exporter>Netbeans XMI Writer</XMI.exporter>
        <XMI.exporterVersion>1.0</XMI.exporterVersion>
      </XMI.documentation>
    </XMI.header>
  - <XMI.content>
    + <Protege.StandardClass xmi.id="Prediction" name="Prediction" abstract="false">
    + <Protege.StandardClass xmi.id="Reduction" name="Reduction" abstract="false">
    + <Protege.StandardClass xmi.id="Empirical Explanations" name="Empirical Explanations" abstract="false">
    + <Protege.StandardClass xmi.id="Organization_Knowledge_Knowwhy" name="Organization_Knowledge_Knowwhy"
        abstract="false">
    + <Protege.StandardClass xmi.id="Empirical Classification" name="Empirical Classification" abstract="false">
    + <Protege.StandardClass xmi.id="Empirical Hypothesis" name="Empirical Hypothesis" abstract="false">
    + <Protege.StandardClass xmi.id="Organization_Knowledge_Knowwhat" name="Organization_Knowledge_Knowwhat"
        abstract="false">
    + <Protege.StandardClass xmi.id="Costing" name="Costing" abstract="false">
    + <Protege.StandardClass xmi.id="Suppliers_Additional Infos" name="Suppliers_Additional Infos" abstract="false">
    + <Protege.StandardClass xmi.id="Vendors and Supply Chain" name="Vendors and Supply Chain" abstract="false">
    + <Protege.StandardClass xmi.id="Purchasing" name="Purchasing" abstract="false">
    + <Protege.StandardClass xmi.id="Sales" name="Sales" abstract="false">
    + <Protege.StandardClass xmi.id="Marketing_strategies" name="Marketing_strategies" abstract="false">
    + <Protege.StandardClass xmi.id="Business Process Knowhow" name="Business Process Knowhow" abstract="false">
    + <Protege.StandardClass xmi.id="Others" name="Others" abstract="false">
    + <Protege.StandardClass xmi.id="Customer Standards" name="Customer Standards" abstract="false">
    + <Protege.StandardClass xmi.id="Customisation" name="Customisation" abstract="false">
    + <Protege.StandardClass xmi.id="Product_specification" name="Product_specification" abstract="false">
    - <Protege.StandardClass xmi.id="OEM_specification" name="OEM_specification" abstract="false">
      - <Protege.StandardInstance.directType>
          <Protege.ExternalClass xmi.idref=":STANDARD-CLASS" />
        </Protege.StandardInstance.directType>
```

```
 - <Protege.StandardClass.directSuperClasses>
     <Protege.StandardClass xmi.idref="Product Design" />
   </Protege.StandardClass.directSuperClasses>
  </Protege.StandardClass>
+ <Protege.StandardClass xmi.id="Tolerance_study" name="Tolerance_study" abstract="false">
+ <Protege.StandardClass xmi.id="Product Design" name="Product Design" abstract="false">
+ <Protege.StandardClass xmi.id="Design_Knowhow" name="Design_Knowhow" abstract="false">
+ <Protege.StandardClass xmi.id="BestPractice" name="BestPractice" abstract="false">
+ <Protege.StandardClass xmi.id="Qualitative_Knowledge" name="Qualitative_Knowledge" abstract="false">
+ <Protege.StandardClass xmi.id="EmployeeTacit_and_Experience" name="EmployeeTacit_and_Experience" abstract="false">
+ <Protege.StandardClass xmi.id="Observation_and_Intuition" name="Observation_and_Intuition" abstract="false">
+ <Protege.StandardClass xmi.id="Written_and_Benchmarking" name="Written_and_Benchmarking" abstract="false">
+ <Protege.StandardClass xmi.id="Quantitative_Knowledge" name="Quantitative_Knowledge" abstract="false">
+ <Protege.StandardClass xmi.id="Manufacturing_Knowhow" name="Manufacturing_Knowhow" abstract="false">
+ <Protege.StandardClass xmi.id="Organization_Knowledge_Knowhow" name="Organization_Knowledge_Knowhow"
   abstract="false">
+ <Protege.StandardClass xmi.id="Organisation Formal Knowledge" name="Organisation Formal Knowledge" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-FUNCTION" name=":JESS-FUNCTION" abstract="false">
+ <Protege.StandardClass xmi.id=":FUNCTION" name=":FUNCTION" abstract="true">
+ <Protege.StandardClass xmi.id=":JESS-RULE" name=":JESS-RULE" abstract="false">
+ <Protege.StandardClass xmi.id=":RULE" name=":RULE" abstract="true">
+ <Protege.StandardClass xmi.id=":JESS-DEFMETHOD" name=":JESS-DEFMETHOD" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-DEFMESSAGE-HANDLER" name=":JESS-DEFMESSAGE-HANDLER" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-DEFFACTS" name=":JESS-DEFFACTS" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-DEFTEMPLATE" name=":JESS-DEFTEMPLATE" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-DEFGLOBAL" name=":JESS-DEFGLOBAL" abstract="false">
+ <Protege.StandardClass xmi.id=":JESS-DEFINITION" name=":JESS-DEFINITION" abstract="true">
+ <Protege.StandardClass xmi.id=":JESS-ENGINE" name=":JESS-ENGINE" abstract="false">
+ <Protege.StandardInstance xmi.id="Jess engine" name="Jess engine">
```

```
+ <Protege.StandardSlot xmi.id="Type_Of_Design_Knowhow" name="Type_Of_Design_Knowhow" valueType="String"
    maxCardinality="1" minCardinality="1">
+ <Protege.StandardSlot xmi.id="Qualitative_KS_Type" name="Qualitative_KS_Type" valueType="String" maxCardinality="1"
    minCardinality="0">
+ <Protege.StandardSlot xmi.id="Qualitative_Knowledge_Statements" name="Qualitative_Knowledge_Statements"
    valueType="String" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id="Quantitative_Knowledge_Statements" name="Quantitative_Knowledge_Statements"
    valueType="String" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id="Quantitative_KS_Type" name="Quantitative_KS_Type" valueType="String" maxCardinality="1"
    minCardinality="1">
+ <Protege.StandardSlot xmi.id="Internal_or_External" name="Internal_or_External" valueType="Symbol" maxCardinality="1"
    minCardinality="1">
+ <Protege.StandardSlot xmi.id="Probability_Factor" name="Probability_Factor" valueType="Symbol" maxCardinality="1"
    minCardinality="1">
+ <Protege.StandardSlot xmi.id="Date_Prepared" name="Date_Prepared" valueType="String" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id="Type_of_Manufacturing_Knowhow" name="Type_of_Manufacturing_Knowhow"
    valueType="String" maxCardinality="1" minCardinality="1">
+ <Protege.StandardSlot xmi.id="Type_of_knowhow" name="Type_of_knowhow" valueType="String" maxCardinality="1"
    minCardinality="1">
+ <Protege.StandardSlot xmi.id="Owner_of_Knowhow" name="Owner_of_Knowhow" valueType="String" maxCardinality="1"
    minCardinality="1">
+ <Protege.StandardSlot xmi.id="Department" name="Department" valueType="String" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id="Organization Name" name="Organization Name" valueType="String" maxCardinality="1"
    minCardinality="0">
+ <Protege.StandardSlot xmi.id=":KB-SAVE" name=":KB-SAVE" valueType="Boolean" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id=":DEFINITION" name=":DEFINITION" valueType="String" maxCardinality="1" minCardinality="0">
+ <Protege.StandardSlot xmi.id=":DEFINITION-NAME" name=":DEFINITION-NAME" valueType="String" maxCardinality="1"
    minCardinality="0">
+ <Protege.StandardSlot xmi.id=":STARTUP-EXPRESSION" name=":STARTUP-EXPRESSION" valueType="String" maxCardinality="1"
    minCardinality="0">
+ <Protege.StandardSlot xmi.id=":STARTUP-FILE" name=":STARTUP-FILE" valueType="String" maxCardinality="1" minCardinality="0">
  <Protege.ExternalSlot xmi.id=":DOCUMENTATION" name=":DOCUMENTATION" />
  <Protege.ExternalClass xmi.id=":THING" name=":THING" />
  <Protege.ExternalClass xmi.id=":STANDARD-SLOT" name=":STANDARD-SLOT" />
  <Protege.ExternalClass xmi.id=":SYSTEM-CLASS" name=":SYSTEM-CLASS" />
  <Protege.ExternalClass xmi.id=":STANDARD-CLASS" name=":STANDARD-CLASS" />
</XMI.content>
</XMI>
```

APPENDIX H – EXECUTION OF WORKFLOW ACTIVITY TASK CONTROLLER

Appendix H displays an example using the workflows function of a PDM system for the novel methodology, Workflow Activity Task Controller (WATC). The appendix illustrates the interactions of various activities which are corresponded to different stages of Figure 7.4 (Section 7.3.1).

Process: WATCv3.7

Category: Default	**Deadline:** --	**Project:** --
State: Executed	**Start Time:** 22 Jan 04 17:48	**End Time:** 22 Jan 04 17:50
Priority: Highest (1)	**Process Initiator:** Administrator	**Template Name:** WATC_test03.7
Primary Object: --		
Description: The purpose of this process is to manage the activities associated with change analysis activities in support of process plan generation and eventually promote to capacity planning		

ID	Activity Name	State	Deadline	Start Time	End Time	Time Until Start	Priority
	Customer_Request	Executed	23 Jan 04 17:48	22 Jan 04 17:48	22 Jan 04 17:48	--	Highest (1)

Description: 1) receive product specification from customer
2) check information

Instructions:

Possible Votes: received

Participant Name	Role	Required	Completed	Vote
Administrator	Administrator	yes	22 Jan 04 17:48	received
--	Creator	no	yes	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Check_&_Add_Customer_Historic_Info	Executed	--	22 Jan 04 17:48	22 Jan 04 17:49	--	Highest (1)

Description: The purpose of this activity is responsible to check customer previous buying records and if necessary add more info on their accounts

Instructions: Review Customers' buying experience, update their account
Choose from the following options if necessary:

1. Invoke the Knowledge-Based System to add all necessary information
2. Assign tasks
3. Inform_Customers to request more information or to CANCEL request

Possible Votes: Activate, Assign, Cancel

Participant Name	Role	Required	Completed	Vote
Administrator	Administrator	yes	22 Jan 04 17:49	Activate

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
Inform_Customer_Comments	inform you	class java.lang.String		--	--
Activate_KBS_Comments	activate	class java.lang.String		--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Activate_KBS	Executed	--	22 Jan 04 17:49	22 Jan 04 17:49	--	Highest (1)

Description:

Instructions: --

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
Inform_Customer_Comments	--	class java.lang.String		--	--
Activate_KBS_Comments	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	class java.lang.String	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	--	--
Windchill_URL	--	class java.lang.String		--	--
ActivateProcessPlanningComments	Start The Capable Aggregate Process Planning System	class java.lang.String	Start The Capable Aggregate Process Planning System	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

NotifyTeam	Executed	--	22 Jan 04 17:49	22 Jan 04 17:49	--	Highest (1)

Description:

Instructions:

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
Administrator	Creator	no	22 Jan 04 17:49	

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Enter_Conceptual_Design_Data	Executed	--	22 Jan 04 17:49	22 Jan 04 17:49	--	Highest (1)

Description: Enter all conceptual data for the design or download the geometric data

Instructions:

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
Administrator	Administrator	yes	22 Jan 04 17:49	

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Engineering

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Windchill_Internal_Functions	Executed	--	22 Jan 04 17:49	22 Jan 04 17:49	--	Highest (1)

Description: to execute the Windchill internal functions such as Check-in Check-out of the Manufacturing-Knowledge XML files in the Document Manager to be readied for the Process-Planning Stage

Instructions: --

Possible Votes: OK, Unavailable, Other Error

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Adding_New_Knowledge	Executed	--	22 Jan 04 17:49	22 Jan 04 17:50	--	Highest (1)

Description:

Instructions:

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
--	Wai M Cheung	no	yes	--
Administrator	Administrator	yes	22 Jan 04 17:50	
Administrator	Creator	no	22 Jan 04 17:49	

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Review_Conceptual_Data	Executed	--	22 Jan 04 17:50	22 Jan 04 17:50	--	Highest (1)	

Description: Produce a new product definition

Instructions:

Possible Votes: Process, Review

Participant Name	Role	Required	Completed	Vote
Administrator	Creator	no	22 Jan 04 17:50	Process

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Activate_Process_Planning	Executed	--	22 Jan 04 17:50	22 Jan 04 17:50	--	Highest (1)	

Description:

Instructions: --

Possible Votes: --

Participant Name	Role	Required	Completed	Vote

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
Inform_Customer_Comments	--	class java.lang.String		--	--
Activate_KBS_Comments	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	class java.lang.String	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	--	--
Windchill_URL	--	class java.lang.String		--	--
ActivateProcessPlanningComments	Start The Capable Aggregate Process Planning System	class java.lang.String	Start The Capable Aggregate Process Planning System	--	--

Category: Default **Responsible Role:** Administrator

Windchill_Internal_Functions	Executed	--	22 Jan 04 17:50	22 Jan 04 17:50	--	Highest (1)

Description: to execute the Windchill internal functions such as Check-in Check-out of the Process-Plan.

Instructions: --

Possible Votes: OK, Unavailable, Other Error

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

CapacityPlanning	Executed	--	22 Jan 04 17:50	22 Jan 04 17:50	--	Highest (1)

Description:

Instructions:

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
--	Wai M Cheung	no	yes	--
Administrator	Administrator	yes	22 Jan 04 17:50	
--	Creator	no	yes	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Inform_customers	Executed	--	22 Jan 04 17:50	22 Jan 04 17:50	--	Highest (1)	

Description: send message to customer to inform them whether their request is being accepted or not

Instructions: --

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Customer Service

Overdue: no

Escalated: no

Iteration: 1

Responsible Role: Administrator

Errors: --

Timer	Not started	--	--	--	--	--

Description:

Instructions: --

Possible Votes: --

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: --

Escalated: --

Responsible Role: --

Errors: --

Review Design	Not started	--	--	--	--	--

Description:

Instructions:

Possible Votes: NotifyTeam, Reject

Participant Name	Role	Required	Completed	Vote
--	--	--	--	--

Variable Name	Value	Type	Default Value	Initialized From	Copied Into
--	--	--	--	--	--

Category: Default

Overdue: --

Escalated: --

Responsible Role: --

Errors: --

Variable Name	Value	Type	Default Value	Initialized From
Inform_Customer_Comments	--	class java.lang.String		--
Activate_KBS_Comments	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	class java.lang.String	1) Invoke the Protege Software Package 2) Run the Organisational Formal Knowledge Based-System 3) Add further knowledge if necessay	--
Windchill_URL	--	class java.lang.String		--
ActivateProcessPlanningComments	Start The Capable Aggregate Process Planning System	class java.lang.String	Start The Capable Aggregate Process Planning System	--

APPENDIX I – IMPLEMENTATION OF XML PARSER

Appendix I depicts the methods of creating a JAVA based XML Parser.

I1: LOAD XML FILE

```
/** Method to load the XML file **/

public Vector read () {
        String url = loadFile();
        //LOAD FILE
        System.out.print ("Loading xml file …");
        Document doc = null;
        try {
                doc = XmlDocument.createXmlDocument(url,true);
        }
        catch(Exception e){e.printStackTrace();}
        System.out.println("ok");
```

I2: walkTree FUNCTION

```
/** Method to find object in Resource Model matching a text string **/

public void walkTree (Resource res, String s) {
        //System.out println ("Checking" + res.Name() + " against " + s);
        if (res.Name().equals(s))
        {
        //System.out println (" Match Found ");
        found_resource = res;
        }
        if (res.Resource().size() > 0) {
                for (int i = 0 ; i < res.Resource().size(); i++) {
                walkTree((Resource)res.Resource().elementAt(i), s);
        }
        }
        }
```

I3: read (resource) FUNCTION EXAMPLE

```
/** Method of read resource function **/

public void read (Resource res) {
        listofknowledgeStatements = new Vector();
        read()
        for (int i = 0 ; i < listofknowledgeStatements.size(); i++) {
        //System.out println ("Working on KS no." + i);
        knowledgeStatement k_temp = (knowledgeStatement)
                listofknowledgeStatements.elementAt(i);
        String mc_temp.getSubjectType();
        found_resource = null;
        walkTree (res, mc_temp)
        if (found_resource !=null) {/* Attach knowledge statement */
                System.out println ("Statement is attached to resource" +
                        found_resource().Name());
                found_resource.addk(k_temp);
        }
        }
        }
```

I4: *get* KNOWLEDGEMENT STATEMENT EXAMPLE

```
/** Begin for Subject Type **/

// Get the "Subject Type" chuck
NodeList listofsubject = (NodeList)firstknowledgeElement.
                        getElementsByTagName("SubjectType");
Node thirdknowledgeNode = (Node)listofsubject.item(0);
if (thirdknowledgeNode.getNodeType() == Node.ELEMENT_NODE)
{
  thirdknowledgeElement = (Element) thirdknowledgeNode;
  System.out println ("Subject Type element instantiated");
  //Get the "subject" attribute from the "subject" chunk
  thirdSubject = thirdknowledgeElement.getAttribute ("subject");
  System.out.println ("Subject = " + thirdSubject);
}
```

I5: Create JAVE OBJECTS

```
// CREATE THE JAVA OBJECT
try
{
  QualitativeKnowledgeStatement  ks = new QualitativeKnowledgeStatement
                                                                    ("default");
  ks.setCreationDate(secondDate);
  ks.setSubjectType(thirdSubject);
  ks.setOwner_organization(XMLorganization);
  ks.setUser(XMLownerknowhow);
  ks.setknowledgeType(XMLKnowledgeType);
  ks.setUserStatement(XMLqualitativeks);
      //System.out.println("toString() from KnowledgeStatement = "+ ks.toString());
      listOfKnowledgeStatements.addElement(ks);
       System.out.println();
      }
    catch (Exception e){e.printStackTrace();
    }
  System.out.println("There are " + listOfKnowledgeStatements.size() + "
knowledge statements in the Vector");
  return listOfKnowledgeStatements;
```

APPENDIX J - RELATED PUBLICATIONS

Appendix J shows a list of twenty-one publications related to this research, five international journals, two book chapters and fourteen conference papers. Of these, eleven were principally written by this author. Over this research period, fourteen papers relating to the project have been presented at various international conferences.

Journal Papers

1. **Cheung, W.M.,** Matthews P.C., Maropoulos, P.G., Gao, J.X., (2007), "Advanced Product Development Integration Architecture – an Out-of-Box solution to support Distributed Production Networks", International Journal of Production Research (IJPR), DOI: 10.1080/00207540601039767.
   Available online at:
   (http://www.informaworld.com/smpp/content~content=a772388959~db=all~order=author)

2. **Cheung, W.M.,** Bramall, D.G., Maropoulos, P.G., Gao, J.X., Aziz, H., (2006). "Organisational Knowledge Encapsulation and Re-use in Collaborative Product Development", *International Journal of Computer Integrated Manufacturing.* Vol. 19 (7), 736 – 750.

3. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.,** (2005). "Open Standard, Open Source and Peer to Peer Tools and Methods for Collaborative Product Development", *Journal of Computers in Industry,* Vol. 56, 260-271.

4. Gao, J.X., Aziz, H., Maropoulos, P.G., **Cheung, W.M.,** (2003). "Application of Product Data Management Technologies for Enterprise Integration", *International Journal of Computer Integrated Manufacturing,* Volume 16 Number 7/8, 491-500.

Journal Paper submitted for review

5. **Cheung, W.M.**, Matthews P.C., Maropoulos, P.G., (2007), "Deployments of Product Lifecycle Management in organisational knowledge to support product development", International Journal of Product Lifecycle Management (IJPLM), Special Issue on: "Exploring the Role of Knowledge Management Technologies in PLM".

Book Chapters

6. Gao, J.X., Aziz, H., Maropoulos, P.G., **Cheung, W.M.,** (2004). "Case study of collaborative knowledge management in the early phases of product development", *Perspectives from Europe and Asia on Engineering Design and Manufacture, A Comparison of Engineering Design and Manufacture in Europe and Asia*, (ed.) Xiu-Tian Yan, Cheng-Yu Jiang and Neal Juster, Kluwer Academic Publishers, ISBN 1-4020-2211-5, 283-295.

7. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.,** (2004). "A design environment for product knowledge management and data exchange", *Methods and Tools for Co-operative and Integrated Design*, (ed.) S. Tichkiewith and D. Brissaud, Kluwer Academic Publishers, 2004, ISBN 1-4020-1889-4, 257-267.

Refereed Conference Papers

8. **Cheung, W.M.** and Maropoulos, P.G., (2006), "A Novel Knowledge Management Methodology to support Collaborative Product Development", Proceedings of the 3rd International CIRP Seminar on Digital Enterprise Technology (DET06), Setúbal, Portugal, 18-20 September 2006.

9. **Cheung, W.M.,** Maropoulos, P.G., Gao, J.X., Aziz, H., (2005). "Ontological approach for Organisational Knowledge Re-use in Product Developing Environments", accepted by, *11th International Conference on Concurrent Enterprising (ICE2005),* University BW Munich, Germany, 20th -22nd June, 87-94.

10. **Cheung, W.M.,** Maropoulos, P.G., Gao, J.X., Aziz, H., (2004). "A Framework for distributed Workflows, Peer-to-Peer and PLM/PDM collaborations to support OEMs and SMEs", *2nd International CIRP Seminar on Digital Enterprise Technology (DET04),* Seattle, Washington USA, 13th – 15th September.

11. **Cheung, W.M.,** Maropoulos, P.G., Gao, J.X., Aziz, H., (2004). "Web-based Knowledge Management Methods for Collaborative Product Development", *2nd International Conference on Manufacturing Research,* Sheffield Hallam University Sheffield,, UK, 1st – 3rd, September, 150-156.

12. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.,** (2003). "Open Standard, Open Source and Peer to Peer Tools and Methods for Collaborative Product Development", *12th Symposium of Product Data Technology EUROPE,* Manchester Conference Centre, UMIST, November, 151-158.

13. **Cheung, W.M.,** Maropoulos, P.G., Gao, J.X., Aziz, H., (2003). "Workflow Activity Task Controller: an Approach to Distribute Knowledge and Information in Collaborative Product Development", *12th Symposium of Product Data Technology EUROPE,* Manchester Conference Centre, UMIST, November 2003, 107-114.

14. **Cheung, W.M.,** Maropoulos, P.G., Gao, J.X., Aziz, H., (2003). "Knowledge-Enriched Product Data Management to Support Aggregate Process Planning", *1st International Conference on Manufacturing Research,* Glasgow, UK, 9th-11th September, 253-258.

15. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.,** (2003). "Tools and techniques for distributed collaborative design and knowledge management", *1st International Conference on Manufacturing Research,* Glasgow, UK, 9th-11th September, 339-344.

16. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.**, (2003). "New Technologies and Standards for Design Knowledge Management in Collaborative Conceptual Design", *9^{th} International Conference of Concurrent Enterprising (ICE2003),* Dipoli Congress Center, Espoo, Finland, 16-18 June.

17. **Cheung, W.M.**, Maropoulos, P.G., Gao, J.X., Aziz, H., (2003). "The Application of Web-based Technologies in Product Data Management and Manufacturing Systems Interoperability and Data Exchange", *10^{th} European Concurrent Engineering Conference*, Plymouth, UK, 14th-16th April, 80-84.

18. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.**, (2003), "A Design Environment for Product Knowledge Management and Data Exchange", *2003 CIRP Design Seminar*, Grenoble, France, 11th-14th May, 43-54.

19. Gao, J.X., Aziz, H., Sharma, R., Welti, M., Bowland, N.W., Maropoulos, P.G., **Cheung, W.M.**, (2002). "Application of Product Data Management Technologies for Enterprise Integration", *1^{st} CIRP International Seminar in Digital Enterprise Technology (DET02),* Durham, UK, 16-17th September, 273-280.

20. Aziz, H., Gao, J.X., Maropoulos, P.G., **Cheung, W.M.**, (2002). "Advanced Tools and Technologies for Collaborative Product Development and Knowledge Management", *18th National Conference on Manufacturing Research (NCMR),* Leeds, UK, September, 115-120.

21. **Cheung, W.M.**, Aziz, H., Maropoulos, P.G., Gao, J.X., (2002). "Integration of a Manufacturing Model with State-of-the-art PDM System", *1^{st} CIRP International Seminar in Digital Enterprise Technology (DET02),* Durham, UK, 16-17th September, 69-72.