Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.
The Design of an Enterprise Information System using
Hierarchical Design Pyramid and Web-based Object Oriented
Model

by
Mr. Kit-Chong Ng

Department of Industrial and Systems Engineering
The Hong Kong Polytechnic University

A thesis submitted to
The Hong Kong Polytechnic University
in accordance with the regulation
for the degree of Doctor of Philosophy

2002
Abstract of thesis entitled

'The Design of an Enterprise Information System using Hierarchical Design Pyramid and Web-based Object Oriented Model'

submitted by Mr. Kit-Chong Ng, Department of Industrial and Systems Engineering

for the degree of Doctor of Philosophy

at The Hong Kong Polytechnic University in 2002.

ABSTRACT

The objective of this research is to design and develop a methodology and architecture of enterprise information system which can be used to capture and integrate global information for knowledge management, and manufacturing planning and control. This new approach improves the overall system modularity, configurability, flexibility and extensibility, which can be used to provide a multitude of new, and lower cost options for managing global supply chain.

This research integrates two widely used concepts of Enterprise Resource Planning (ERP) and Business Process Reengineering (BPR). Literature review and an industrial survey were conducted to examine and identify the reasons why the ERP system has not been implemented successfully. The author attempts to overcome its weakness using an innovative design and implementation methodology. As a result of this research, a conceptual design model called the Hierarchical Design Pyramid (HDP) which can be used to model ERP under an enterprise reengineering (BPR) context is developed. This model encompasses structural and object-oriented tools for
the development of a global manufacturing information system. Furthermore, to
provide a systematic framework for the implementation of this model through Internet,
an innovative approach called Web-based Object-Oriented Model (WOOM) is also
proposed. The HDP and WOOM models developed by the author have contributed to
the design and development of a sophisticated, but easy to use ERP system, it
provides a total integrated solution for all areas of a global enterprise information
system through the Internet. In the implementation of this global information system
architecture, four major domains have been developed which are logic domain,
presentation domain, service domain, and database domain. Distributed object
technology such as inheritance and polymorphism are used for data model, system
and programming development.

The theory and methodologies proposed in this research are validated with the help of
a collaborated company in the Teaching Company Scheme where the author was
employed as teaching company associate for three years from 1996 to 1999. Six
months after implementing the prototype system, the collaborated company averaged
twice the number of orders received over the Internet as the original system. The
results of this research provide a better understanding and an innovative approach to
the design and development of a global enterprise information system. As a result of
this successful research project, the author set up his own Internet software business
in 1999 and received an accolade from the “Ten Outstanding Young Digi Persons” in
2000 organized by Hong Kong Productivity Council, Hong Kong Junior Chamber.
and Information Technology Services Department of HKSARG.
ACKNOWLEDGEMENTS

The author expresses his gratitude to Dr. W.H. Ip of the Hong Kong Polytechnic University, and to Prof. T.C. Lee of the Hong Kong Polytechnic University, who were directly involved in and supported this project.

Special thanks are due to Dr. Henry Tseng of Kingtronics Industrial Co., Ltd., the teaching company scheme collaborated company, for his many useful technical discussions and recommendations.
# TABLE OF CONTENTS

ABSTRACT .................................................................................................. I

ACKNOWLEDGEMENTS .......................................................................... III

TABLE OF CONTENTS ............................................................................ IV

LIST OF FIGURES .................................................................................... VII

GLOSSARY ................................................................................................ IX

1. INTRODUCTION .................................................................................. 2
   1.1 Research Background .................................................................... 2
   1.2 Research Methodology .................................................................. 5
   1.3 Identification Of The Problem Area .............................................. 6
   1.4 Research Objectives .................................................................... 8
   1.5 Overview Of The Thesis ................................................................. 9

2. LITERATURE REVIEWS ..................................................................... 10
   2.1 Manufacturing Resources Planning (MRP II) .............................. 11
   2.2 Enterprise Resource Planning (ERP) .......................................... 14
   2.3 Design Methodologies .................................................................. 19
      2.3.1 Structural approach ............................................................... 19
      2.3.2 Object-oriented paradigm ..................................................... 23
   2.4 Enterprise Integration ................................................................... 27
   2.5 Digital Nervous System (DNS) ..................................................... 31
   2.6 Summary ..................................................................................... 37

3. INDUSTRIAL SURVEY ....................................................................... 42
   3.1 Survey Objective .......................................................................... 42
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2</td>
<td>Survey Methodology</td>
<td>43</td>
</tr>
<tr>
<td>3.3</td>
<td>Data Analysis</td>
<td>44</td>
</tr>
<tr>
<td>3.4</td>
<td>Summary</td>
<td>52</td>
</tr>
<tr>
<td>4.</td>
<td>DESIGN METHODOLOGY</td>
<td>54</td>
</tr>
<tr>
<td>4.1</td>
<td>Hierarchical Design Pyramid (HDP) – Three Steps Approach</td>
<td>54</td>
</tr>
<tr>
<td>4.2</td>
<td>Step 1: Business Process Reengineering</td>
<td>60</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Fundamental</td>
<td>63</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Process</td>
<td>63</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Radical</td>
<td>63</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Dramatic</td>
<td>64</td>
</tr>
<tr>
<td>4.3</td>
<td>Step 2: Structural Model Design</td>
<td>64</td>
</tr>
<tr>
<td>4.3.1</td>
<td>IDEF design</td>
<td>65</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Quality Walk Through Procedure</td>
<td>71</td>
</tr>
<tr>
<td>4.4</td>
<td>Step 3: Object-Oriented Design And Analysis</td>
<td>75</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Object model</td>
<td>75</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Formulation of hierarchical static model</td>
<td>77</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Formulation of ladder dynamic model</td>
<td>80</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Inheritance</td>
<td>82</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Polymorphism</td>
<td>84</td>
</tr>
<tr>
<td>5.</td>
<td>IMPLEMENTATION APPROACH</td>
<td>87</td>
</tr>
<tr>
<td>5.1</td>
<td>Web-Based Object-Oriented Model (WOOM)</td>
<td>87</td>
</tr>
<tr>
<td>5.2</td>
<td>Model Overview</td>
<td>88</td>
</tr>
<tr>
<td>5.3</td>
<td>Logic Domain (LD)</td>
<td>93</td>
</tr>
<tr>
<td>5.4</td>
<td>Presentation Domain (PD)</td>
<td>97</td>
</tr>
<tr>
<td>5.5</td>
<td>Service Domain (SD)</td>
<td>99</td>
</tr>
<tr>
<td>5.6</td>
<td>Database Domain (DD)</td>
<td>100</td>
</tr>
<tr>
<td>6.</td>
<td>TEACHING COMPANY RESEARCH PROJECT</td>
<td>104</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>104</td>
</tr>
<tr>
<td>6.2</td>
<td>Overview</td>
<td>105</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Project timetable</td>
<td>105</td>
</tr>
<tr>
<td>6.2.2</td>
<td>The collaborated company</td>
<td>106</td>
</tr>
</tbody>
</table>
6.3 Statement Of The Problem ......................................................................................... 107

6.4 Design And Analysis – HDP ..................................................................................... 108
  6.4.1 Business process reengineering ......................................................................... 109
  6.4.2 Structural analysis - IDEF model ......................................................................... 118
  6.4.3 Object model design ............................................................................................. 128

6.5 Implementation - WOOM ......................................................................................... 133
  6.5.1 Physical architecture ............................................................................................. 133
  6.5.2 Logical architecture ............................................................................................... 134
  6.5.3 WOOM architecture ............................................................................................... 135
  6.5.4 Graphical user interface ........................................................................................ 142

6.6 Implementation Considerations .................................................................................. 145
  6.6.1 The situation before and after ERP ....................................................................... 146
  6.6.2 Project team and training ...................................................................................... 147
  6.6.3 Change management and leadership ..................................................................... 150
  6.6.4 A good implementation methodology .................................................................. 152

7. RESULTS AND DISCUSSION ..................................................................................... 154
  7.1 Analysis Of Results ................................................................................................. 154
  7.2 Industrial Adaptability ............................................................................................. 156
  7.3 Limitations Of The Methodology ............................................................................. 158
  7.4 Benefits Of The Methodology .................................................................................. 160

8. CONCLUSIONS AND FUTURE WORK ..................................................................... 165
  8.1 A Conclusion Of The Research Work Undertaken .................................................... 165
  8.2 Contribution From Teaching Company Scheme ...................................................... 168

  8.3 Future Works ............................................................................................................ 170
  8.3.1 Integration with Real-Time Monitoring System .................................................. 170
  8.3.2 Web security ......................................................................................................... 173

  8.4 Statement Of Originality And Contribution To Knowledge .................................... 174

REFERENCES .................................................................................................................. 176

APPENDIX I – QUESTIONNAIRE .................................................................................... 193

APPENDIX II – SAMPLE SOURCE CODE ..................................................................... 197

APPENDIX III – DATABASE SCHEMA ............................................................................. 207
LIST OF FIGURES

Figure 2.1: Major areas examined in the literature review ........................................... 10
Figure 2.2: A typical MRP II system .............................................................................. 12
Figure 2.3: Digital Nervous Network (DNS) (source: Gates, 1999) ................................ 34
Figure 2.4: Summary of worldwide projects using various technologies for enterprise
information technology ................................................................................................. 41
Figure 3.1: Company that used computerized manufacturing information system ........... 45
Figure 3.2: Software systems implemented in the company .......................................... 46
Figure 3.3: Companies that have adopted Internet-based information system ............... 46
Figure 3.4: Internet-based information system will be adopted over next two years ......... 47
Figure 3.5: Adoption of Internet-based information system by manufacturing sector. .......... 47
Figure 3.6: Adoption of Internet-based information system by manufacturing type ....... 48
Figure 3.7: Adoption of Internet-based information system by company size ............... 49
Figure 3.8: Limitations in the application of IT in the company ..................................... 50
Figure 3.9: Important changes in the application of IT in the company .......................... 51
Figure 3.10: Essential elements for improving the efficiency of software design .......... 51
Figure 4.1: Hierarchical Design Pyramid ...................................................................... 56
Figure 4.2: IDEF0 methodology ..................................................................................... 58
Figure 4.3: Three-step approach to Hierarchical Design Pyramid ............................... 59
Figure 4.4: IDEF0 general model representation ............................................................ 65
Figure 4.5: Node tree of the generic ERP model from the HDP approach ....................... 66
Figure 4.6: Node A0 building a generic ERP model ....................................................... 67
Figure 4.7: The decomposed page from the block A0 of the generic ERP model ........... 68
Figure 4.8: Class relationship diagram of item master module ..................................... 78
Figure 4.9: Class relationship diagram of sales and purchasing system ....................... 79
Figure 4.10: Dynamic object diagram of purchasing an item ....................................... 81
Figure 4.11: Ladder diagram of purchasing an item ...................................................... 81
Figure 4.12: Multiple inheritances ................................................................................ 83
Figure 5.1: A single WOOM object ............................................................................. 90
Figure 5.2: Interaction between WOOM objects .......................................................... 91
Figure 5.3: Logic domain ............................................................................................. 94
Figure 5.4: Hierarchical object model ......................................................................... 95
Figure 5.5: Constitution of logic domain ..................................................................... 96
Figure 5.6. Presentation domain .................................................................................. 97
Figure 5.7: Constitution of presentation domain .......................................................... 98
Figure 5.8: Constitution of service domain ................................................................. 100
Figure 5.9: The logical flow of transactions ................................................................. 101
Figure 5.10: Constitution of service domain .............................................................. 103
Figure 6.1: Teaching company research timetable ...................................................... 105
Figure 6.2: Global strategic/manufacturing operation .................................................. 110
Figure 6.3: Global manufacturing strategy development and implementation ............ 111
Figure 6.4: Development of the web-based ERP system .............................................. 120
Figure 6.5: Initiated ERP design ................................................................................ 121
Figure 6.6: Design ERP functions ............................................................................. 122
Figure 6.7: "Object model of "People" and "Customer" ............................................... 128
Figure 6.8: Database schema of "People" and "Customer" .......................................... 129
Figure 6.9: Ladder diagram of user authentication ................................................... 130
**GLOSSARY**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASE</td>
<td>Computer Aided Software Engineering</td>
</tr>
<tr>
<td>CIM</td>
<td>Computer Integrated Manufacturing</td>
</tr>
<tr>
<td>CORBA</td>
<td>Common Object Request Broker Architecture</td>
</tr>
<tr>
<td>DDE</td>
<td>Dynamic Data Exchange</td>
</tr>
<tr>
<td>DFD</td>
<td>Data Flow Diagram</td>
</tr>
<tr>
<td>DOOD</td>
<td>Distributed Object-Oriented Database</td>
</tr>
<tr>
<td>EDI</td>
<td>Electronic Data Interchange</td>
</tr>
<tr>
<td>EI</td>
<td>Enterprise Integration</td>
</tr>
<tr>
<td>ERP</td>
<td>Enterprise Resource Planning</td>
</tr>
<tr>
<td>HDP</td>
<td>Hierarchical Design Pyramid</td>
</tr>
<tr>
<td>IDEF</td>
<td>Integration Definition Methodology</td>
</tr>
<tr>
<td>JIT</td>
<td>Just In Time</td>
</tr>
<tr>
<td>MES</td>
<td>Manufacturing Execution System</td>
</tr>
<tr>
<td>MPS</td>
<td>Master Production Schedule</td>
</tr>
<tr>
<td>MRP</td>
<td>Material Requirement Planning</td>
</tr>
<tr>
<td>MRP II</td>
<td>Manufacturing Resource Planning</td>
</tr>
<tr>
<td>OLE</td>
<td>Object Linking and Embedding</td>
</tr>
<tr>
<td>OO</td>
<td>Object-Oriented</td>
</tr>
<tr>
<td>OOAD</td>
<td>Object-Oriented Analysis and Design</td>
</tr>
<tr>
<td>OOD</td>
<td>Object-oriented Design</td>
</tr>
<tr>
<td>PLC</td>
<td>Programmable Logic Controller</td>
</tr>
<tr>
<td>SA&amp;D</td>
<td>Structured Analysis and Design</td>
</tr>
<tr>
<td>STD</td>
<td>State Transition Diagram</td>
</tr>
<tr>
<td>TCS</td>
<td>Teaching Company Scheme</td>
</tr>
<tr>
<td>WOOM</td>
<td>Web-based Object-Oriented Model</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 Research Background

Over the last decade we have seen the transformation of Hong Kong into a global manufacturing information centre. Hong Kong manufacturers have not only moved into China, but have also relocated to Malaysia, Thailand, Vietnam and Mauritius. Advances in information technology allow manufacturers to optimize production operations by choosing the best sites around the world at which to carry out specific operation. In global manufacturing, a nervous center in Hong Kong with sophisticated information infrastructure has to be set up to direct the flow of information, materials and commands.

This evolution can be traced back to the liberalization of the People’s Republic of China (PRC) in 1978, and since then the entire landscape of Hong Kong manufacturing has experienced drastic changes. The manufacturing sector has been the largest contributor to the gross domestic product (GDP) of Hong Kong until the early 1980s. The traditional mode of operation of many Hong Kong manufacturing firms was to conduct all the manufacturing processes locally under the same roof. In the 1980s, in a drive towards enhanced competitiveness and in response to the market opening of the Mainland of China (the Mainland), many manufacturers in Hong Kong began to shift their land and labour intensive operations to the Mainland and other low-cost economies in the region. Nevertheless, they usually retained the headquarters control and other high value-added business functions of the firms in Hong Kong. Such expansion in scale of operation and re-organisation of business
processes were made possible by the advance in information technology and telecommunications.

It is estimated that Hong Kong-based enterprises employ some five million workers in the Pearl River Delta and hundreds of thousands elsewhere in the region. Hong Kong is the control centre of a regional-wide production network as well as an international and regional services hub. Over the past decades, Hong Kong has gradually transformed into a high-skilled, higher value-added economy. However, a recent survey found that the Hong Kong manufactures were significantly less likely to have invested in new technology and labor-saving machinery (Chiu et al., 1997). Another study also found that Hong Kong manufacturers were lagging behind other Asian manufacturers in the field of electronic commerce and manufacturing information technology (Kurt Salmon 1996). The major reasons of the low utilization rate of information technology in Hong Kong manufacturing industry may be due to the inflexibility of the existing information system architectures and the awareness of the local manufacturers in using information technology to streamline their operations.

Some researches have studied the dispensation of information technology in manufacturing, for example, two comprehensive architectures, the holonic manufacturing system (HMS) (Brussel et al., 1998; Valckenaers et al., 1994; Christensen et al., 1994) and the agent-oriented manufacturing system (Kwok & Norrie, 1993; Kouiss et al., 1995) have been presented in recent years as the basis for manufacturing systems; however, they cannot be considered to be a particular type of solution specialized for global manufacturing. On the other hand, enterprise information system has become an extremely active research area, with agent
technology being recognized as a promising approach for its implementation (Guttman et al., 1998). Agent-based approaches have been widely applied for enterprise integration, manufacturing production planning, scheduling, process control, and material handling in the past decade (Shen and Norrie, 1999), but few significant research results have been reported for adopting agent technology for supply chain management, particularly for global manufacturing enterprises.

Agent-oriented approaches to enterprise modeling include a data flow based approach (Huntbach, Jennings, and Ringwood, 1995), the workflow-based methodology proposed by Kendall et al. (Kendall, et al., 1997) for complex distributed manufacturing systems, multi-agent approaches to the dynamic scheduling of flexible manufacturing system (Kouiss et al., 1995) and the CIMOSA architecture-based system for implementation and control of manufacturing systems (Rabelo and Camarinha-Matos, 1995).

In the design paradigm, the common techniques, such as the state transition diagram (STD), have been widely used for conversation protocol specification due to its simplicity. The STD is purely concerned with modeling the behavior of a single thread of execution. The weakness is that it does not reflect the asynchronous character of the underlying communication (Martial, 1992). Furthermore, it is not easy to represent integration of protocols. Similarity, finite-state machines (FSM) model parallelism only in terms of states and transitions, with no attempt to model the interactions between the parallel threads. When FSMs are used to represent the allowable speech act sequences in conversations, their advantage is that they are easy to conceptualize and implement, and may be adequate for the routine interaction of
many kinds of simple agents (Bradshaw, et al., 1997). However, they have limited ability to express many kinds of constraints relevant to conversations (for example, information about overall timing and security). Dooley Graph (Parunak, 1996) is an alternative formalism for visualizing agent interrelationships within a conversation. All in all, the methodologies from design to implementation have received relatively limited attention in the previous research. As such it was one of the motivations behind this research study.

The focus of this research is to realize a novel architecture of enterprise information system to capture and integrate global information into knowledge management, business operations, and commerce, with a particular focus on design methodology and implementation approach. There have been numerous definitions and methodologies formulated for enterprise information system which feature in a comprehensive review presented in Chapter 2.

1.2 Research Methodology

Traditional research methodologies such as literature review and quantitative survey is regarded as the basic research methodologies. Detailed analysis of existing applications and industrial packages were also adopted as a benchmarking of existing information system for global manufacturing. The research with the collaboration company is considered as a valuable experience. Most importantly, the collaboration company has provided an actual industrial environment for the validation of proposed conceptual design and implementation methodology so as to strengthen the research value, the deliverables and their contributions to this field.
In the past, structured analysis and design is the most common methodology to study Enterprise Resources Planning (ERP) and other information systems. Conventional analysis and design of an information system involves the identification of data structures, information flow, task decomposition, etc. However, research in software engineering and intuitive science show that structured model might not be the best methodology for understanding a system. This is simply because human does not think in this way (West 1996). Therefore, many people moved to object-oriented approach during the 1990’s. OOAD (Object oriented analysis and design) becomes a new stream for system analysis and design. In OOAD, real objects in the system and the relationships between objects are identified. This is much closer to the human way of thinking than traditional analysis techniques. The object-oriented paradigm promotes reuse ability, reduce development time, and improve analysis, design and software quality. The cost and the risk that is associated with software development are reduced. The introduction of object-oriented technology brings simplification and better understanding of requirements analysis. This leads to easy, logical design and actual implementation.

In this research project, a hybrid approach that encompassed structural methodology and Object-Oriented Analysis and Design methodology has been employed as the primary analysis and design tools during system development life cycle.

1.3 Identification Of The Problem Area

The speed at which a company anticipates and adapts to market forces is a critical source of competitive advantage for global manufacturers. Customers today are demanding more – more information, faster delivery, and superior pre- and post-sales
service. They expect their suppliers to be flexible enough to respond to their needs on multiple dimensions. A company's ability to seamlessly integrate all of its supply chain elements – from departments and business units to suppliers and customers on a global scale – will determine success in the 21st Century.

Market leaders have already squeezed excess costs out of their supply chains by replacing legacy systems with Enterprise Resource Planning (ERP) systems. Growth-minded executives are now shifting their focus from cost cutting to tightening the links with suppliers and customers. They are partnering with their customers early in the product development process to ensure that product and service requirements are met. The leaders are also leveraging Internet technologies and adding functionality to ERP applications to capture and integrate key customer information into strategic planning.

However, many manufacturers pointed out that the existing global enterprise information systems are extremely primitive. The principle objective of this research is therefore to contribute to the advancement of architecture of enterprise information system to capture and integrate global information into knowledge management, business operations, and commerce, with a particular focus on design methodology and implementation approach. The work focuses on the identification of architecture and framework coherently integrates the modeling, design and implementation of an Internet-based ERP system. Methodologies are proposed and validated for implementing the underlying system so as to enhance the overall system modularity, configurability, flexibility and extensibility and to provide a multitude of new, lower cost options for integrating with global supply chain partners.
In response to research background in Section 1.1, the secondary goal of this research project is to construct using a systematic procedure of a prototype system of web-based ERP following the proposed design methodology and implementation approach. Finally, the prototype system has been implemented and tested in the collaborated company so as to validate the proposed concepts and methodologies.

1.4 Research Objectives

In this research, the objectives are summarized as follows:

(i) To identify the importance of global information technologies and their associated implementation considerations from the view of manufacturers and system designers through an industrial research with the collaborated company;

(ii) To integrate two widely used concepts of Enterprise Resource Planning (ERP) and Business Process Reengineering (BPR);

(iii) To develop a framework for the design and implementation of an enterprise information system for global manufacturing;

(iv) To design a methodology and architecture of enterprise information system, which can be used to capture and integrate global information for knowledge management, and manufacturing planning and control;

(v) To implement the data model and system development; and
(vi) To validate the proposed methodology in the collaborated company.

1.5 Overview Of The Thesis

This thesis is organized in eight chapters. Chapter one gives an introduction to the problem and the problem area is identified. Literature reviews on previous researches and latest technologies are provided in Chapter two. Chapter three presents an industrial survey with an analysis of the data collected from the industrial survey. The proposed design methodology for the global enterprise information system is presented in Chapter four. Chapter five describes the approach to implement the proposed design methodology. The teaching company research project is presented in Chapter six. Research results and the discussion of advantages and limitations of the proposed methodology are stated in Chapter seven. Chapter eight concludes works and results in this research, and suggests the future development in this area. Finally, references and appendixes are attached at the end of this thesis.
2. LITERATURE REVIEWS

Five major areas have been investigated in the literature review including, manufacturing resource planning (MRP II), enterprise resource planning (ERP), existing design methodologies, enterprise integration, and digital nervous system (DNS). Figure 2.1 illustrates the schematic of the literature review.

![Diagram showing Design Methodologies, Manufacturing Resource Planning (MRP II), Enterprise Resource Planning (ERP), and Digital Nervous System (DNS) connected by arrows indicating a flow or relationship.]

Figure 2.1: Major areas examined in the literature review

The review on MRP II aims to identify problems in the design and implementation of a standard manufacturing software system. The discussion on ERP systems emphasis on their new features based on the state-of-the-art information technologies. Existing design methodologies are then described and their inadequacies are elaborated. The review on enterprise integration illustrates the requirements in the development of global manufacturing information technology. Literatures on digital nervous system
demonstrate the future technology to support the information infrastructure of enterprise. Finally, hierarchical design pyramid (HDP) is proposed as the core design methodology and web-based object oriented model (WOOM) is implemented to solve the problem identified in Chapter 1.

2.1 Manufacturing Resources Planning (MRP II)

Material Requirement Planning (MRP) is a calculation technique for planning purchase orders and manufacturing orders. MRP II is an extension of MRP. By linking other business management applications to MRP, MRP II becomes a method for effective planning of all resources of a manufacturing company. MRP II is a systematic approach to plan production in complex multistage manufacturing systems. Most of the manufacturing application software is based on the concept of MRP II as the central application. MRP II is the dominant application software system for today's manufacturing management (Turbide, 1995).

The major inputs of MRP II are the master production schedule (MPS), the product structure records and the inventory status records. The MPS is a statement of the orders for finished goods. It indicates the quantity of each finished good that needs to be produced by a time period. The MPS usually involves the purchase order released from the customer and the work order to be issued to the shop floor. The product structure records or the bill of material (BOM) is a list of components and material that are required to produce a finished good. It contains information on the relationships of parts, components and assemblies. It also shows the quantity of each component and the sequence to assemble a finished good. The inventory status
records contain the data of on-hand inventory, inventory location and on-order inventory.

The outputs of a MRP II system include purchase orders for raw material, work orders for shop floors, reschedule notices for job review, and capacity requirements for labor arrangement. Figure 2.2 illustrates a typical MRP II system.

![Diagram of MRP II system]

Figure 2.2: A typical MRP II system

The study of MRP II has received a lot of attention in the field of manufacturing management. Many researchers (Woodgate, 1989) (Villa, 1990) (Poon, 1991) (Muller, Jackman, and Fitzwater, 1990) have dealt with defining MRP II and integrating MRP II with new functions and other systems. In recent years, the execution and feedback phase has been described as a weakness in many MRP II systems (Murthy and Ma, 1991). The lack of well-defined data structure and the piecemeal development approach of traditional MRP II systems had posed a number of problems (Correll, 1995). The following limitations inherent in the traditional
design and development approach are identified:

(i) Inventory data is not synchronized in different countries.

(ii) Data does not represent an abstraction of reality.

(iii) Pending data affects estimation.

(iv) Real-time data is not available.

(v) Production data is not taken into account.

On the other hand, the manufacturing technology, modes of production and hardware devices have advanced dramatically over the past several decades; the network and computer systems have got faster and more standardized (Loose, 1995). Unfortunately, the manufacturing software has not kept up with the changes in hardware. This problem is usually related to the quality of software. Inflexible software makes the whole system very difficult or even impossible to modify. Unsupported software and hardware are a great barrier for the growing of MRP II applications (Deloitte Consulting, 1998). Moreover, task specific MRP type applications do not support linkage to other enterprise information systems for data sharing. Therefore, a new technology for developing flexible and scalable manufacturing information system is required to support the modern manufacturing enterprises.
2.2 Enterprise Resource Planning (ERP)

Enterprise Resources Planning (ERP) is an enterprise-wide computer management system. Davenport (1993) described ERP as a commercial software package that promises the seamless integration of all the information flowing through the company — financial, human resources, supply chain, and customer information. ERP represents the application of the latest Information Technology (IT) to the Manufacturing Resources Planning (MRP/MRPII) system. MRP was born in the 1960s from a joint research and development effort between two innovators: J.I. Case (Tom and Elden, 1996), the manufacturer of construction machinery and its partner IBM. In the 1990’s MRP was further extended to cover all the major business functions and is called ERP. In addition to the new business functions, it has included many technology innovations including the move to relational database management systems (RDBMS), the use of a graphical user interface (GUI), open systems and a client / server architecture (Robert, 1996).

In the early 1960s, computer hardware vendors from worldwide intended to develop a structured methodology for scheduling and planning materials for manufacturing industry. Their effort to develop computer software to assist manufacturing enterprises was one of the strategies to increase the sales of hardware. Today, ERP is one of the fastest growing segments of the software market. AMR (1999), a Boston-based research and consulting firm predicted an annual growth of more than 37 percent through the year 2000. As ERP functionality increases, more industries will consider implementation imperative in order to maintain the competitive edge. SAP, PeopleSoft, J.D. Edwards, Oracle and Baan are the predominant vendors of ERP.
software market (Thibodeau, 1998). Currently SAP has the largest market share with
the largest number of installations.

The evolution of ERP industry is closely related to technological enhancements in the
hardware industry. Initial MRP solutions were big, clumsy and expensive. These
applications were disjointed and slow and cost millions of dollars for the installation
and implementation (Loizos, 1998). Manufacturers required a lot of technical staffs
to support the rigid mainframe computers on which they ran, and therefore were
accessible only to large companies belong to Fortune 500.

In the 1970s, some MRP applications, described in Section 2.1, started to become
modularized. In addition, critical business functions such as cost accounting and
Capacity Requirements Planning (CRP) were integrated into MRP to establish the
next generation of software (Glass, 1998). The application that was used mainly by
the manufacturing and operations departments was extend to satisfy the needs of the
entire company's operating system. In the vendors market, software companies were
formed to take an increasing share from the large hardware vendors, and competition
enhanced the product features and technology.

In the 1980s, the technological advancements brought the most rapid acceleration to
this industry by the advent of the "downsizing" trend. The midrange computer market
brought the price and performance of the hardware into the hands of Fortune 1000 to
2000 companies. On this platform, the software matured and companies competed to
add functionality. At the same time, the personal computer was bringing computer
hardware and software into just about any company. The proliferation of spreadsheets
like Lotus 123 put crude but simple and manageable computer tools into the hands of the small to mid-sized manufacturers who could not afford large data processing staff and equipment. The mid-80s saw the birth of PC-based manufacturing systems that could replicate the MRP and standard manufacturing support features available previously only from large software vendors.

One of the major technological innovations for the PCs was their linking together into Local Area Networks (LAN) so that information could be shared within the organization rather than shared between individual hard drives and floppy disks. The late '80s and early '90s saw tremendous technical advancements from Intel and Microsoft, as well as the growth of UNIX-based operating environments where this became viable technology for medium-sized and smaller companies.

The 1990s accelerated the downsizing trend to the point that Client/Server computing is the worldwide standard among information technology departments. The other technical advancement created an uproar in the well-established manufacturing software market; the faster, more powerful PCs allowed the proliferation of the Graphical User Interface (GUI), primarily through the Windows and Windows 95 operating systems. This forced the entire software industry to either enhance or totally rewrite their products to supply their users with the same easy-to-use and intuitive interface as the rest of their desktop (O’Brien, 1999). As manufacturing software packages scrambled to satisfy the marketplace, the companies that spent the better part of a decade heaping functionality into their legacy products have had the greatest difficulty getting new product to market, and therefore losing market share in the process.
The second major influence of this decade has indirectly been the result of technology because of the expansion of the global business marketplace. As companies expanded their markets, and, consequently, their supply chain from vendors to customers, there has been an increased emphasis on localization of products with multilanguage, multicurrency and international standards support (Hammer, 1999). Along with the expanding international requirements, companies did not want their MRP systems to facilitate only the operations groups, but to serve the entire business enterprise. The enterprise (hence ERP) demanded integration of Quality, Human Resources and Payroll systems, as well as the distillation of the voluminous data generated in these systems into a format those executives and line-workers can understand using EIS (Executive Information Systems) and OLAP (On Line Analytical Processing) tools.

Dr. Mark L. Spearman, University Chair of Manufacturing Management at the Culverhouse College of Commerce and Business Administration at The University of Alabama, said that while the potential of the Internet was being proven, Enterprise Resources Planning software was taking industry by storm (Spearman and Hopp, 2000). ERP is the continuation of the development of computer software for managing operations that began in the 1960s with Manufacturing Requirements Planning (MRP). MRP represented a very simple model that could perform elementary inventory planning functions but which is not suited to a capacitated manufacturing environment. Consequently, several generations of computer-aided production planning have followed. These included Manufacturing Resources Planning (MRP II), Closed-Loop MRP, Business Resources Planning (BRP) and, most recently, Enterprise Resources Planning (ERP).
Spearman and Hopp said ERP systems are huge, costing hundreds of millions of dollars and taking years to implement. "Recently they have become so complex that corporations have been willing to modify their business practices to fit the software in order to speed installation. While ERP has seen some success, the overall picture has been disappointing," he said.

A recent survey of 63 companies revealed a return on investment of a negative $1.5 million. There are two reasons for this. One, ERP performs basic logistical functions using exactly the same logic of the MRP of the 1960s. While simple processing was required given the computing resources of the 1960s, it is inexcusable today. Instead of re-engineering the entire paradigm, ERP vendors have provided multiple add-ons in the form of what are called ‘Advanced Planning Systems’ (Spearman and Hopp, 2000)

Moreover, the ERP systems are based on monolithic, database-centric architectures where the database models have become extremely complex to accommodate diverse uses of data in an enterprise-wide system. This architecture becomes a barrier as the ERP vendors attempt to ‘componentize’ their systems and scale and distribute them across the World Wide Web. Worse, there are no fundamental principles for enterprise control that guide the vendors in where to break the system into components that can integrate as a seamless whole (Spearman and Hopp, 2000).

Although ERP represents the application of the state-of-the-art-technology to manufacturing application software, with the accelerating advancement of Internet technology, recent researchers identify many areas of improvement in ERP
applications (Coburn, 2000). Researchers such as Scott (1998) state that making use of Internet-based technology in manufacturing applications will be the trend of the 21st century. Other researchers illustrate the application of component technology using object-orientated models to design and implement ERP system for quicker, faster implementation and customization (Booch, Rumbaugh, and Jacobson, 1999).

Nevertheless, all the researchers have not mentioned a systematic design and implement methodology to achieve a highly customizable, flexible, scaleable web-based ERP system, which is in-line with the enterprise-wide business strategy and planning. This study is therefore aimed to contribute to the development and understanding of a new methodology to ERP design and development.

2.3 Design Methodologies

Sections 2.1 and 2.2 have reviewed the state-of-the-art development of MRPII and ERP systems. This section gives a detailed elaboration of the design approach of global enterprise information system.

2.3.1 Structural approach

Several structural analysis methodologies are available for the design and modelling of ERP system. Some examples are SADT (Structured Analysis and Design Technique) by Douglas T. Ross of SoftTech (Ross 1977, 1985), SSADM (Structured System Analysis and Design Methodology) by Learmonth and Burchett Management Systems and the Central Computer and Telecommunications Agency (Cutts 1991), SASS (Structural Analysis and System Specification) (DeMarco, 1979; Gane and
Sarson, 1979), DFD (Data Flow Diagram) approach (DeMarco 1979, Gane and Sarson 1979), and STD (State-Transition Diagram) (Martial, 1992). In general, structural methods such as SSADM and STD are generally inadequate for problem domains analysis with natural concurrence.

On the other hand, Integrated DEFineition methodology (IDEF), derived from the US Air Force Integrated Computer Aided Manufacturing (ICAM) (Wisnosky 1987) initiative, is a structured analysis and design method based on graphic and text descriptions of functions, information and data. The IDEF method is widely understood and well documented (Hill 1995, US Air Force) and the method originally defined by the ‘Architects Manual’ includes guidance for modelling, together with rules for model syntax, diagram and model format and text presentation, as well as structured model validation, document control procedures and interview techniques. A key concept of IDEF modelling is the definition of a ‘Context’ and the modeller’s ‘Viewpoint’ to establish an explicit common understanding of the boundary and aspect of the system being modelled.

The IDEF method is an expansion of SADT associated with the ICAM model (integrated computer-aided manufacturing) for manufacturing system study. This method allows us to model the functional structure of an organisation with IDEF0, its information structure with IDEF1, and its dynamic structure with IDEF2 (KBSI, 1999).

The primary strength of IDEF0 is that the method has proven effective in detailing the system activities for function modeling, the original structured analysis
communication goal for IDEF0. Activities can be described by their inputs, outputs, controls, and mechanisms (ICOMs). Additionally, the description of the activities of a system can be easily refined into greater and greater detail until the model is as descriptive as necessary for the decision-making task at hand. The hierarchical nature of IDEF0 facilitates the ability to construct (AS-IS) models that have a top-down representation and interpretation, but which are based on a bottom-up analysis process. Beginning with raw data (generally interview results with domain experts), the modeler starts grouping together activities that are closely related or functionally similar. Through this grouping process, the hierarchy emerges. If an enterprise's functional architecture is being designed (often referred to as TO-BE modeling), top-down construction is usually more appropriate. Beginning with the top-most activity, the TO-BE enterprise can be described via a logical decomposition. The process can be continued recursively to the desired level of detail. When an existing enterprise is being analyzed and modeled (often referred to as AS-IS modeling), observed activities can be described and then combined into a higher level activity. This process also continues until the highest level activity has been described (CAM.I, 1980).

In fact, one of the observed problems with IDEF0 models is that they often are so concise that they are understandable only if the reader is a domain expert or has participated in the model development. One problem with IDEF0 is the tendency of IDEF0 models to be interpreted as representing a sequence of activities. While IDEF0 is intended not to be used for modeling activity sequences, it is easy to do so. The activities may be placed in a left to right sequence within decomposition and connected with the flows. It is natural to order the activities left to right because, if
one activity outputs a concept that is used as input by another activity, drawing the activity boxes and concept connections is clearer. Thus, without intent, activity sequencing can be imbedded in the IDEF0 model. In cases where activity sequences are not included in the model, readers of the model may be tempted to add such an interpretation. This anomalous situation could be considered a weakness of IDEF0. However, to correct it would result in the corruption of the basic principles on which IDEF0 is based and hence would lose the proven benefits of the method. The abstraction away from timing, sequencing, and decision logic allows concision in an IDEF0 model. However, such abstraction also contributes to comprehension difficulties among readers outside the domain. This particular problem has been addressed by the object-orient methodology.

In term of the operation, the first step in IDEF modelling is thus concerned with establishing the objectives of the modelling effort from which a context and viewpoint can evolve. Moreover, this is a top-down method which starts from general applications and moves on to more specific issues, from a single page that represents an entire system to more detailed pages that explain how the subsections of the system work. It includes both procedure and a language for constructing a model of the decisions, actions, and activities in an organisation. In this research, “DESIGN/IDEF”(1994) software package was selected to be the modelling tool to build the functional ERP model. Based on this methodology, a set of procedures was generated and the required documents and forms were also produced to match these procedures. Furthermore, the model can be used to compare the available ERP software packages and the functional feasibility studies can be identified. As a result,
an ERP system which best fits a company can be produced using the structured approach.

2.3.2 Object-oriented paradigm

More recently, object-oriented methodologies have tried to regroup data and functions in the same independent entity: the object. Since 1980, with OOD (Booch, 1994) many object-oriented methodologies have been developed: HOOD (HOOD User Group, 1992) in 1986, OOSA (Shlaer and Mellor 1988) in 1988, OOA (Coad, 1991), OMT (Rumbaugh et al., 1991), which seems to be appreciated in industry: M*OBJECT (Berio et al., 1995).

Because of the independent nature of objects, these methodologies favour knowledge specification, and facilitate coherence, reuse and evolution; they enable us to model complex system. With object-oriented paradigm, system is defined in terms of objects. The object-oriented method represents a model of a system that is based on real world entities.

Objects are the basic elements in object-oriented paradigm. An object represents a real-world entity, such as a car, form, etc. Each object is complete and self-contained with well-defined attributes and operations. In traditional software development approaches, operations determine the structure of the system. However, in object-oriented system, operations and data combine to form objects. Structure of the system is determined by a set of objects instead of operations.
The characteristics of the object-oriented paradigm are data abstraction, encapsulation, inheritance, and polymorphism (Booch, 1994). They are summarized as follows:

(i) Data abstraction is a way to group objects with similar operations (methods) and attributes (data) together. An object’s method and attributes are determined by its class. A class is defined as the type of a specified object and the object is an instance of its own type. For example, purchase order (object) is a real world object and purchase order is a subset of document. Since all objects of the same class have common internal attributes and external operations, all of them can be accessed by a set of methods in the same way.

(ii) Encapsulation is a combination of operations and attributes. It means that an object contains both a set of attributes and a set of operations that can be performed on those attributes. An object can only be accessed though the methods those are predefined for external use. External objects are prevented from applying unsuitable methods to the other objects, and each object’s internal attributes and private methods are hidden. Hence, encapsulation enables an extensive data abstraction, enhances data security, and facilitates application consistency.

(iii) Inheritance is the ability to associate attributes that are common to all instances of a class of objects. It means that subclass of objects can inherit those attributes of its parent class. The aim of inheritance is to reduce as much work as is absolutely necessary to define new object types. The inheritance
mechanism introduces reuse of methods defined in generic objects and provides a powerful and easily extensible system.

(iv) Polymorphism provides a generic mechanism to perform an operation on different objects. When an operation is performed on an object, the actual operation is determined by the type of object and the operation addresses. This approach generalizes behaviour over many types of objects and allows a higher degree of abstraction in system development. It also separates the design of object operations from the actual implementation.

A recent research stress that the multi-agent systems paradigm promises to be a valuable software engineering abstraction for the development of computer systems (Roberto and Flores-Mendez, 1999). Basically, agents and objects share some characteristics that sometimes make them hard to differentiate. For example, agent-oriented programming (AOP) could be considered a specialization of the object-oriented programming (OOP) paradigm (Shoham 1997). On the one hand, OOP views systems as consisting of objects communicating with one another to perform internal computations, whereas AOP specializes this view to have agents (instead of objects), whose internal computations are based on beliefs, capabilities, choices and so forth, and which communicate with each other using messages adopted from speech-act theory.

Although this conception allows one to appreciate the similarities between agent and objects, their differences are less obvious. There are three main differences that have been identified between agents and objects (Jennings et al., 1998):
(i) The first is in the degree to which agents and objects are autonomous. It thus does not think of agents as invoking methods upon one-another, but rather as requesting actions to be performed. In the object-oriented case, the decision lies with the object that invokes the method. In the agent case, the decision lies with the agent that receives the request.

(ii) The second important distinction is with respect to the notion of flexible (reactive, pro-active, social) autonomous behaviour.

(iii) The third important distinction is that agents are each considered to have their own thread of control in the standard object model, there is a single thread of control in the system.

A review of agent theories, architectures and languages can be found in (Wooldridge and Jennings, 1995). A survey of cooperative environments for engineering design can be found in (Shen and Barthès, 1996), a review of some simultaneous engineering systems in (Molina et al., 1995), and a taxonomy for multi-agent robotics in (Jenkin et al., 1996).

Eventually, it is relatively difficult to identify agents or objects, and this approach is rather too abstract in regard to user requirement. There is a need for mechanisms for finding, using, presenting, managing and updating object services and information. This study is therefore intended to develop an innovative approach to encompass the structural and object-oriented design approaches in the context of enterprise integration. Section 2.4 reviews the background and the development of enterprise
integration as related to the various design methodologies. A new concept – digital nervous system (DNS) – will also be discussed in Section 2.5 in order to introduce the rationale of the proposed design and implementation methodologies in this study.

2.4 Enterprise Integration

A comprehensive review on design methodologies has been conducted in Section 2.3. In order to design a sophisticated global enterprise information system, the adoption of novel methodology is extremely essential. This section gives a detailed discussion on the requirement of enterprise integration for global manufacturing to support the proposed design methodology.

Global competition and rapidly changing customer requirements are forcing major changes in the production styles and configuration of manufacturing organizations (Weston and Contts, 1994). Increasingly, traditional centralized and sequential manufacturing planning, scheduling, and control mechanisms are being found insufficiently flexible to respond to changing production styles and highly dynamic variations in product requirements. The traditional approaches limit the expandability and reconfiguration capabilities of the manufacturing systems. The traditional centralized hierarchical organization may also result in much of the system being shut down by a single point of failure, as well as plan fragility and increased response overheads. Therefore, the need of enterprise integration is rising.

Enterprise integration is the discipline that connects and integrates business and manufacturing systems so the right people and systems have the right information at the right time. It provides a natural way to overcome the mentioned problems, and to
design and implement an enterprise information infrastructure. Recently, a number of researchers (Jennings et al., 1995; Jennings, Sycara and Wooldridge, 1998) have attempted to apply agent technology to manufacturing enterprise integration. Agent technology has been considered as one of the approaches for developing industrial distributed systems.

Enterprise Integration comprises all the activities necessary to ensure that the manufacturing enterprise functions as a coordinated whole, by itself and within extended enterprise, and includes the following:

(i) A set of strategies, architectures, concepts, and values guiding ongoing business and product-related processes, practiced by the enterprise's entire workforce, and aligned with specific performance objective.

(ii) Enablers used in the well-managed physical, financial, human resources, and information infrastructures that bind the processes together.

Regarding the information systems, enterprise integration is defined as the extent to which various information systems are formally linked for the sharing of consistent information within an enterprise. It can be conceptualized along two dimensions: (a) data integration, and (b) enterprise communication networking (Marshak, 1991). Enterprise communication networking has been represented along two dimensions: connectivity, and flexibility.
A recent research (NGM, 1997) point out that enterprise integration should be comprised of:

(i) Shared strategic direction for the organization itself, consistently deployed at all levels;

(ii) The integration of both internal functional organizations and external partners and suppliers; the integration of end-to-end processes that cross functional and organizational boundaries;

(iii) The establishment of a cooperative culture throughout the organization and the empowerment of people;

(iv) Integration of financial assets and human resources;

(v) The standardization of data and the sharing of corporate information through a common information infrastructure; and

(vi) The integration of an organization's physical assets to ensure a flexible and adaptive physical infrastructure.

Enterprise integration efforts represent an understanding of the potential of global competition for product development and product enhancement in both defense and commercial enterprises. The current vertically developed productivity programs, such as computer-aided acquisition and logistic support (CALS), concurrent
engineering/simultaneous product and process engineering, product data specifications/national plan for intelligent product definition, total quality management (TQM), and rapid acquisition of manufactured parts (RAMP) could benefit from enterprise integration (NGM, 1997).

The enterprise integration concept provides a new vision for design, production, and support of manufacturing systems. Enterprise integration is a new approach to integrate a variety of product and process information and automation productivity technology initiatives under one system integration umbrella. It can provide the basis for a national initiative for inter/intra enterprise system integration by assisting in national consensus building, stimulating the development of commercial products, and developing effective and realistic tools, techniques, and methods for the business enterprise.

Widespread use of computer-aided design and engineering (CAD/CAE) has created a new environment in which product description data are becoming available in digital form to support a wide range of manufacturing applications. Computer-aided acquisition and logistic support and concurrent engineering are industry initiatives to enable and accelerate the use of integration of digital technical information for system acquisition, design, manufacture, and support. Related programs such as total quality management should be included in the vision for enterprise integration (Schael, 1997). A comprehensive strategy has been developed to facilitate the transition from the current paper-intensive mode of operations to a highly automated and integrated mode, thereby substantially improving productivity and quality of the system acquisition and logistic support process. Implementation of the enterprise integration is already
underway and is leading to a major impact of how industry conducts business (Kosanke and Nell, 1997).

Enterprise integration also provides a unique opportunity to achieve major productivity and quality improvements through carefully planned and managed investment. Initially, the changes will be gradual as building blocks are put in place and specific portions of the enterprise life cycle are enhanced.

As the cumulative impact of enterprise integration and infrastructure modernization are realized in industry, more far-reaching changes will occur in the way functions are accomplished, leading to additional major savings. Enterprise integration implementation will result in a lower system life cycle cost, shortened product acquisition/development times, and improvements in reliability, maintainability, and quality. In order to benefit from the advantages of enterprises integration, the information system architecture is considered to be a key driver. This research project, therefore, aims to identify and propose a novel architecture for the design and development approach. The next section will discuss the rationale of a total integrated system – digital nervous system – for global enterprises in view of design and implementation aspects.

2.5 Digital Nervous System (DNS)

Comprehensive reviews on MRPII and ERP have been presented in Sections 2.1 and 2.2. This section will further introduce an extended concept – Digital Nervous System (DNS). National networks are essential today to major manufacturers' processes: global networks will be absolutely essential for manufacturing enterprises.
Increasing requirements for process flexibility, and customer demand for an increasing variety of product and option variation, exacerbate manufacturing complexity and will drive the need for global networks.

As information and information exchange become more valuable to economic performance, those countries that develop an effective advanced information infrastructure will gain competitive advantage in global markets. Instead of just chasing low wages, as has been the trend in the recent past, manufacturers increasingly will choose to locate and invest in countries whose infrastructure is able to handle rapid and efficient information control and dissemination and the integration of diverse business operation. Hong Kong is a sophisticated nerve center for global manufacturing.

With such a global network, digital technology radically increases the speed at which business is conducted. It changes the relationship between businesses and their customers, and it transforms the roles of individual workers. How well an organization adapts to this changing business landscape will depend on the health of its internal digital processes—its "digital nervous system." (Gates 1999)

A biological nervous system playing in living organisms controls the basic systems—respiratory, circulatory, and digestive—that make life possible. It also receives sensory stimuli, transmits them to the brain, and instantly triggers a response. In humans the nervous system makes it possible to think and plan with foresight and creativity. Similarly, a digital nervous system performs alike functions within an enterprise. It allows a company's internal processes to operate smoothly and quickly.
A DNS enables an organization to respond to customer feedback quickly, gives it the ability to react to its competitive environment in a timely manner, and empowers employees with critical knowledge. The key is how effectively an organization manages the flow of its digital information. All kinds of information—numbers, text, audio, video—can now be put into digital form. Advanced hardware and software has also made it possible for enterprises of all sizes to reshape the way they conduct their business. In fact, many companies have already made many of the necessary investments in information technology to allow employees to obtain, store, share, and act upon information in new ways, from corporate intranets to the global network of the Internet.

In order to achieve this, enterprises create digital processes that manage and deliver important information to their employees who need it, helping them to respond effectively to changes and new opportunities, make decisions quickly, and fix problems as soon as they arise. Under the pressure of globalization, the digital nervous system must enable three or more individuals in very different countries and cultures to work together in a virtual work cell – at short notice and on a critical, short-term project – as if they were long-time team members in a single location and from a single culture. To achieve this, the system must (Ng and Ip, 2001):

(i) Transport high volumes of data at rates that can support real-time interventions and with high integrity and security.

(ii) Present just enough information to each individual, just in time and in a language- and culture-sensitive form useful to the individual.
(iii) Facilitates cooperative work among geographically and culturally separated individuals.

(iv) Quickly and seamlessly integrate people and intelligent machines into smoothly functioning systems.

With the above system features, digital processes can improve organizational intelligence by giving workers the information necessary to be more effective. Nowadays, there is no standard digital nervous system because each manufacturing enterprise is different. According to Gates’ (1999) definition, a digital nervous system (Figure 2.3) must transform three major elements of any business:

![Digital Nervous System Diagram](image)

Figure 2.3: Digital Nervous Network (DNS) (source: Gates, 1999)

(i) Information flow and relationships among employees within an enterprise (knowledge management);
(ii) Internal business processes (business operations); and

(iii) Relationships to customers and business partners (commerce).

A key element of a digital nervous system is linking these different systems—knowledge management, business operations, and commerce—together. To make digital information flow an intrinsic part of an enterprise, the followings are twelve essential steps defined by previous research (Gates, 1999):

For knowledge management:

(i) Insist that communication flow through the enterprise over e-mail so that everyone can act on news with the highest speed.

(ii) Study sales data online to find patterns and share insights easily. Understand overall trends and personalize service for individual customers.

(iii) Use PCs for business analysis, and shift knowledge workers into high-level thinking work about products, services, and profitability.

(iv) Use digital tools to create cross-departmental virtual teams that can share knowledge and build on each other's ideas in real time, worldwide. Use digital systems to capture corporate history for use by anyone.
(v) Convert every paper process to a digital process, eliminating administrative bottlenecks and freeing knowledge workers for more important tasks.

For business operations:

(vi) Use digital tools to eliminate single-task jobs or change them into value-added jobs that use the skills of a knowledge worker.

(vii) Create a digital feedback loop to improve the efficiency of physical processes and improve the quality of the products and services created. Every employee should be able to easily track all the key metrics.

(viii) Use digital systems to route customer complaints immediately to the people who can improve a product or service.

(ix) Use digital communications to redefine the nature of your business and the boundaries around your business. Become larger and more substantial or smaller and more intimate as the customer situation warrants.

For commerce:

(x) Trade information for time. Decrease cycle time by using digital transactions with all suppliers and partners, and transform every business process into just-in-time delivery.
(xi) User digital delivery of sales and service to eliminate the middleman from customer transactions.

(xii) Use digital tools to help customers solve problems for themselves, and reserve personal contact to respond to complex, high-value customer needs.

To cope with the vision of digital nervous system, this research project is focused on identifying systematic methodology that can help enterprises develop customized solutions in these three areas.

### 2.6 Summary

Today, manufacturers are moving towards open architectures for integrating their activities with those of their suppliers, customers and partners within global manufacturing networks. This chapter reviews the literature on design and development technologies, which provide various ways to design and implement the enterprise information system. Figure 2.4 summarizes some research studies in this area.

<table>
<thead>
<tr>
<th>Project</th>
<th>Research Group</th>
<th>Domain</th>
<th>Main characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIMS</td>
<td>Park et al 1993 Lockheed Martin</td>
<td>Agile Manufacturing</td>
<td>Using the Internet</td>
</tr>
<tr>
<td>ATP</td>
<td>Nist 1998 NIST</td>
<td>Agile Manufacturing</td>
<td>Plug-and-play framework</td>
</tr>
<tr>
<td>N/A</td>
<td>Pancerella et al 1995 Sandia Lab</td>
<td>Agile Manufacturing</td>
<td>An agent defined as an autonomous, encapsulated software component</td>
</tr>
<tr>
<td>N/A</td>
<td>Wunderli et al 1996 ETH Zentrum</td>
<td>CIM Systems</td>
<td>Database agents for CIM systems</td>
</tr>
<tr>
<td>Madefast</td>
<td>Collaborative Engineering</td>
<td>Using the Internet</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>MADEsmart</td>
<td>Jha et al 1998 Boeing</td>
<td>Wrapper agents for legacy system encapsulation</td>
<td></td>
</tr>
<tr>
<td>iDCSS</td>
<td>Klein 1995 CSS, MIT</td>
<td>A integrated model that combines existing coordination technologies</td>
<td></td>
</tr>
<tr>
<td>ABMA</td>
<td>Budenske et al 1998</td>
<td>Middleware architecture</td>
<td></td>
</tr>
<tr>
<td>CIIMPLEX</td>
<td>Peng et al 1998 UMBC</td>
<td>Service agents (Name Server, Facilitator Agent, Gateway Agent)</td>
<td></td>
</tr>
<tr>
<td>CLAIM</td>
<td>Malkoun &amp; Kendall 1997 RMIT</td>
<td>Methodology for Enterprise Integration using agents</td>
<td></td>
</tr>
<tr>
<td>IA framework</td>
<td>Pan &amp; Tenenbaum 1991 EIT</td>
<td>Large number of computerized assistants, known as Intelligent Agents (IAs)</td>
<td></td>
</tr>
<tr>
<td>AMACOIA</td>
<td>Sprumont &amp; Muller 1996 U. of Neuchatel</td>
<td>Using simulated annealing to search problem space</td>
<td></td>
</tr>
<tr>
<td>IAO</td>
<td>Kwok and Norrie 1994 U. of Calgary</td>
<td>A rule based object system for developing intelligent manufacturing software</td>
<td></td>
</tr>
<tr>
<td>MetaMorph I</td>
<td>Maturana &amp; Norrie 1996 U. of Calgary</td>
<td>Mediator-centric architecture; dynamic clustering &amp; cloning; learning</td>
<td></td>
</tr>
<tr>
<td>MetaMorph II</td>
<td>Shen et al 1998 U of Calgary</td>
<td>Hybrid architecture; Mediators as subsystem coordinators and interfaces to the main system</td>
<td></td>
</tr>
<tr>
<td>N/A</td>
<td>Liu &amp; Sycara 1994, 1995 CMU</td>
<td>CP&amp;CR (Constraint Partition and Coordinated Reaction) for constraint satisfaction; Anchor&amp;Ascend for distributed constraint optimization</td>
<td></td>
</tr>
<tr>
<td>N/A</td>
<td>Fischer 1994 DFKI</td>
<td>Hierarchical layered architecture</td>
<td></td>
</tr>
<tr>
<td>CAMPS</td>
<td>Miyashita 1998</td>
<td>Repair-based methodology together with constraint-based mechanism</td>
<td></td>
</tr>
<tr>
<td>N/A</td>
<td>Hasegawa et al 1994 Toshiba</td>
<td>Using HMS approach</td>
<td></td>
</tr>
<tr>
<td>ABACUS</td>
<td>McElenery et al 1998 UCB, UMIST</td>
<td>Manufacturing Scheduling</td>
<td>Using functional agents; BDI approach for agent design</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------</td>
<td>--------------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td>ADDYMS</td>
<td>Butler &amp; Ohtsubo 1992</td>
<td>Manufacturing Scheduling</td>
<td>Agents represent physical resources; dynamic local resource scheduling</td>
</tr>
<tr>
<td>AMC</td>
<td>Goldsmith &amp; Interrante 1998 Sandia Lab</td>
<td>Manufacturing Scheduling</td>
<td>Using physical agents: part agents and machine agents</td>
</tr>
<tr>
<td>CORTES</td>
<td>Sadeh &amp; Fox 1989, Sycara et al 1991 CMU</td>
<td>Manufacturing Scheduling</td>
<td>Micro-opportunistic techniques for solving scheduling problems</td>
</tr>
<tr>
<td>DAS</td>
<td>Burke &amp; Prosser 1994 U. of Strathclyde</td>
<td>Manufacturing Scheduling</td>
<td>Hierarchical architecture with agents representing resources, resource groups, and a whole scheduling process</td>
</tr>
<tr>
<td>LMS</td>
<td>Fordyce &amp; Sullivan 1994</td>
<td>Manufacturing Scheduling</td>
<td>Using functional agents; voting protocol for communication</td>
</tr>
<tr>
<td>Sensible Agents</td>
<td>Barber et al 1998 U of Texas at Austin</td>
<td>Manufacturing Scheduling</td>
<td>Implemented as CORBA objects communicating through ILU object environment</td>
</tr>
<tr>
<td>N/A</td>
<td>Kouiss et al 1997</td>
<td>Manufacturing Scheduling</td>
<td>Each agent represents a work center</td>
</tr>
<tr>
<td>N/A</td>
<td>Murthy et al 1997</td>
<td>Manufacturing Scheduling</td>
<td>A-team architecture</td>
</tr>
<tr>
<td>N/A</td>
<td>Sousa &amp; Ramos 1997 ISEP/IPP</td>
<td>Manufacturing Scheduling</td>
<td>Using HMS approach; dynamic scheduling</td>
</tr>
<tr>
<td>AARIA</td>
<td>Parunak et al 1998 ITI, U of Cincinnati</td>
<td>Manufacturing Scheduling</td>
<td>Using autonomous agents to represent physical entities</td>
</tr>
<tr>
<td>IFCF</td>
<td>Lin and Solberg 1992 Purdue</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Resource agents represent physical resources; market-like control model</td>
</tr>
<tr>
<td>MASCADA</td>
<td>Bruckner et al 1998 Daimler-Benz AG, KULeuven</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Emergent Behavior in Manufacturing Control; Proactive Disturbance Handling; Hot Plugable Agents</td>
</tr>
<tr>
<td>MASCOT</td>
<td>Parunak 1993 ITI</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>A shared ontology &amp; a base set of realistic modules</td>
</tr>
<tr>
<td>Reagere</td>
<td>Berry &amp; Kumura 1998 Penn State U.</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Based on blackboard architecture</td>
</tr>
<tr>
<td>SFA</td>
<td>Parunak 1996 NCMS</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Real manufacturing applications</td>
</tr>
<tr>
<td>YAMS</td>
<td>Parunak 1987 ITI</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>One of earliest applications in the domain</td>
</tr>
<tr>
<td>------</td>
<td>-----------------</td>
<td>----------------------------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>N/A</td>
<td>Baker 1991 U. of Cincinnati</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Market-Driven Contract Net; forward &amp; backward scheduling</td>
</tr>
<tr>
<td>N/A</td>
<td>Duffie &amp; Piper 1986 Wisconsin</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Agents represent physical resources, parts, and humans; part-oriented scheduling</td>
</tr>
<tr>
<td>N/A</td>
<td>Interrante &amp; Goldsmith 1998 Sandia Lab</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Type-A, Type-B and Type-C agents</td>
</tr>
<tr>
<td>N/A</td>
<td>Saad et al 1995 Vanderbilt</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Production Reservation approach; machine-centered &amp; part-centered negotiation</td>
</tr>
<tr>
<td>N/A</td>
<td>Ouelhadj et al 1998 U. of Toulouse 1</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Agents represent physical resources</td>
</tr>
<tr>
<td>N/A</td>
<td>Patriti et al 1997, Schaefer et al 1996</td>
<td>Manufacturing Scheduling</td>
<td>Layered architecture; using different mechanisms</td>
</tr>
<tr>
<td>N/A</td>
<td>Tseng et al 1997 HKUST</td>
<td>Manufacturing Scheduling &amp; Control</td>
<td>Market-like model for manufacturing control with agents representing resources</td>
</tr>
<tr>
<td>N/A</td>
<td>Fleury et al 1996</td>
<td>Manufacturing System Optimization</td>
<td>'Triple coupling' of multi-agent techniques, simulated annealing, and simulation</td>
</tr>
<tr>
<td>MAPP</td>
<td>Hayes 1998 U. of Minnesota</td>
<td>Process Planning</td>
<td>Combination of sequential and blackboard architectures</td>
</tr>
<tr>
<td>N/A</td>
<td>Yan et al 1998 Leipzig</td>
<td>Project Management</td>
<td>Using mobile agents</td>
</tr>
<tr>
<td>ARMOSE</td>
<td>Overgaard et al 1994 Odense U.</td>
<td>Robotics</td>
<td>Each joint of a robot is modeled as an agent</td>
</tr>
<tr>
<td>N/A</td>
<td>Choi and Park 1997</td>
<td>Shop Floor Scheduling of Shipbuilding Yard</td>
<td>An economical method for developing intelligent agent systems</td>
</tr>
<tr>
<td>ADE</td>
<td>Mehra &amp; Nissen 1998 Gensym Co.</td>
<td>Supply Chain Management</td>
<td>Using delegation based event handling similar to JavaBeans</td>
</tr>
<tr>
<td>ISCM</td>
<td>Fox et al 1993 U. of Toronto</td>
<td>Supply Chain Management</td>
<td>Agent Building Shell (ABS); Coordination Language (COOL); functional agents</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>N/A</td>
<td>Brugali et al 1998 Politecnico di Torino</td>
<td>Supply Chain Management</td>
<td>Using mobile agents to an industrial process</td>
</tr>
<tr>
<td>N/A</td>
<td>Swaminathan et al 1996 CMU</td>
<td>Supply Chain Management</td>
<td>Supply chain library with structural elements (agents) and control elements for coordination</td>
</tr>
<tr>
<td>KRAFT</td>
<td>Gray et al 1998 KRAFT consortium</td>
<td>Transformation and Reuse of Knowledge</td>
<td>Mediators as knowledge brokers</td>
</tr>
<tr>
<td>N/A</td>
<td>Papaioannou &amp; Edwards 1998 Loughborough</td>
<td>Virtual Enterprise</td>
<td>Using mobile agents</td>
</tr>
</tbody>
</table>

Figure 2.4: Summary of worldwide projects using various technologies for enterprise information technology

Chapters 4 and 5 will be based on the finding in this literature review to design and develop the enterprise information system to enhance the overall system modularity, configurability, flexibility and extensibility, which can be used to provide a multitude of new and lower cost options for managing with global manufacturing.
3. INDUSTRIAL SURVEY

3.1 Survey Objective

A quantitative survey to review the practice of global information technology in Hong Kong manufacturing sector was performed in this research. The survey was prepared and organized by the Hong Kong Polytechnic University and the collaborated company – Kingtronics Industrial Co., Ltd. under the Teaching Company Scheme project in which the author was the teaching company associate.

The purposes of this survey were:

(i) to identify the importance of global information technologies.

(ii) to examine the appropriateness or otherwise of current information system design and implementation methodology.

(iii) to explore the extent and concern of system implementation considerations from the view of manufacturers and system designers.

The following sections describe the research methodology as well as the data analysis and results. Survey results will be presented in graphical format to help the explanations. A summary is provided in Section 3.4 to conclude the industrial survey and the extent to which the survey objectives have been accomplished.
3.2 Survey Methodology

In order to formulate this survey research, in 1997, the author initially conducted a series of studies with the collaborated company to identify general concept on the usage and implementation of global information technology in Hong Kong manufacturing industry. Drawing upon this study, a focus group consisting of representatives from the various functional departments in the collaborative company was held; the author developed a conceptual model, which helped to guide the major research effort of the study, both in the design of a survey instrument and in selection of the targeted respondents in all 1,000 business partners associated with the company.

The survey was conducted by the author and was carried out in the 3rd quarter of 1998. One thousand manufacturers in Hong Kong were randomly selected from the customer and vendor databases of the Kingtronics Industrial Co., Ltd. Most of the targeted respondents have production plants in Mainland China. The mode of survey was by mail and followed up by telephone interviews. All of the telephone interviews were used to clarify the unreadable handwriting and any ambiguous answers. The telephone interview asked questions mainly about the ranking questions (number 8 – 10). The interviews usually lasted not more than fifteen minutes and were not tape-recorded. Immediately following each telephone interview, the author wrote an extended summary from notes taken during the interview. The author usually included quotes or near-quotes in the summary but did not work from a recording or transcript of the interview.

The target respondents of this survey were the top management and senior engineers of Hong Kong’s small-to-medium-size manufacturers. Data collected from the
questionnaire survey were aimed at the managing director and/or I.T. director who are responsible for the technology implementation strategy within the company.

The survey questionnaire contained fifteen (15) questions in total. These were divided into three sections covering the (a) Current Information Technology Management, (b) Deployment of Global Information Technology, and (c) Background Information. It was designed to be closed and multiple-choice questions. The survey instrument can be seen in Appendix I. The total number of questionnaires distributed was 1,000. The number of responses was 187, or 18.7%. The overall response rate to the survey was acceptable and represented a reasonable sample underpinned the usage of information technology in manufacturing industry.

The data drawn from the questionnaires was analyzed very carefully and compiled into an extensive database. The major results were summarized and presented in the following sections.

3.3 Data Analysis

According to the survey methodology developed in Section 3.2, the survey data collected were analyzed and presented in graphical formats to ease explanation and elaboration.

From Figure 3.1, it is shown that around 72% of manufacturers have adopted manufacturing information system, e.g., CAD/CAM, material requirement planning, product delivery, purchase order processing, quality assurance and control, forecasting, sales order processing, inventory management, shop floor management.
The result shows that computerized manufacturing information system is essential to most of the manufacturers in Hong Kong.

Company that used computerized manufacturing information system

![Pie chart showing percentages of manufacturers using computerized manufacturing information system.]

□ Yes □ No □ Don't know

Figure 3.1: Company that used computerized manufacturing information system.

Figure 3.2 depicts the present situation of adoption of information system in manufacturing. The results reveal that the top three popular information systems in manufacturing companies are CAD/CAM, inventory management and accounting software. The penetration rates are 87%, 67%, and 45% respectively. On the other hand, shop floor control and material requirement planning are the least successful applications in the sample group. The penetration rates are only 3% and 8% respectively. The results demonstrate that manufacturers are eager to adopt new technologies in their operations but there were various difficulties encountered during the implementation process. The following analysis further elaborates this observation.
Figure 3.2: Software systems implemented in the company.

From the figure, it can be seen that 30% of the manufacturers have adopted Internet-based information system and half of them have not yet adopted this kind of system (see Figure 3.3), however, 80% of them would implement such system over next two years (see Figure 3.4).

Figure 3.3: Companies that have adopted Internet-based information system.
Figure 3.4: Internet-based information system will be adopted over next two years.

The results show that most of the manufacturers are aware of using Internet-based information system to streamline their global manufacturing operations.

In this survey, the correlation of manufacturing sectors and adoption of Internet-based information system has been investigated (see Figure 3.5).

Figure 3.5: Adoption of Internet-based information system by manufacturing sector.
It is observed that around 52% of manufacturers in electronics industry have already adopted Internet-based information system. Over 30% of respondent in toys and plastics industries have implemented such system. The relatively lower adoption rate can be found in metal and clothing industries at 17% and 11% respectively. On the other hand, the adoption rate of Internet-based information system in electronics sector is relatively high. One of the possible explanations is because of the demand of real-time, on-line information in the electronics sector.

Figure 3.6 illustrates the level of adoption of Internet-based information system by manufacturing type.

![Adoption of Internet-based Information System by Manufacturing Type](image)

Figure 3.6: Adoption of Internet-based information system by manufacturing type.

In this survey, four categories have been identified, namely Assembly, Process, Project and Job Shop. From the result, we observed that the highest adoption rate of Internet-based information system is job shop typed manufacturers (around 44%). Around 43% of manufacturers in project type have also implemented the Internet-
based information system. Both of these two manufacturing types require highly flexibility in project management; the communication cost would be a critical factor for them to consider in selecting application software architecture, therefore Internet-based system may be their choice.

![Adoption of Internet-based information system by company size](image)

Figure 3.7: Adoption of Internet-based information system by company size.

Figure 3.7 depicts that around 52% of companies who have adopted Internet-based information system are large-sized company. Large-sized company is defined as the company with over 500 staff. Medium-sized company is defined as the company with 51 to 500 staff, while small-sized company is defined as the company with less than 50 staff. Around 29% of companies who have adopted Internet-based information system are medium-sized companies while only 19% of them are small-size companies. It can be deduced that large companies would have more resources in investing information technology, especially Internet-based information system.
Figure 3.8: Limitations in the application of IT in the company.

Around 38% of the respondents felt that a complicated system review procedure is the major limitation in the application of IT in their companies. Besides, 28% of manufacturers consider that a rigid project structure make I.T. enhancement difficult. The lack of staff to investigate and manage new technologies is also considered as a substantial limitation (see Figure 3.8). It also shows that a systematic and flexible system design and architecture will be a catalyst for the manufacturers to apply IT in their organizations.

Over 90% of manufacturers consider the three factors – (a) New software development methodology, (b) Review and change company policy, and (c) Modular development – are the most important changes in the application of IT in their companies (see Figure 3.9).
Figure 3.9: Important changes in the application of IT in the company.

Figure 3.10: Essential elements for improving the efficiency of software design.

In the survey, five different importance levels have been identified, from one ("1") to five ("5") - five is the highest. Figure 3.10 illustrates that modular system architecture is the most important element for improving the system design and development cycle.
Besides, design and development methodology, and integration framework are also important to system design while the development tool is considered to be the least important factor for improving efficiency of software design. Clearly, the result of survey shows that manufacturers are well aware of the need of a novel software development methodology.

3.4 Summary

The survey results form a rich database that provides the analysis of various aspects of computerization of manufacturing industry in Hong Kong; adaptation level of Internet-based information system in various industry sectors; considerations in new system implementation; and difficulties encountered in manufacturing software design. We have not attempted to explore all of these questions in detail but the principal results of the survey can be summarized as follows:

(i) Most of the manufacturers are aware of using Internet-based information system to streamline their operations in HK and China.

(ii) Manufacturers in HK are aware of the need of a novel software development methodology.

(iii) A systematic and flexible system design will be a catalyst for the manufacturers to apply IT in their organizations.

(iv) Small and medium size companies will encounter more difficulties in information system implementation.
(v) The low utilization rate of information technology in manufacturing industry is due to the lack of a comprehensive and easy-to-adopt system development approach.

All in all, the methodology and results of this survey is an important step to identify the need of this research, it enhances our motivation of the research and helps to establish the objectives described in Section 1.1. Moreover, based on the finding of this survey, the company’s initiative and commitment, and literature reviews - a new design methodology for global manufacturing system has been proposed and will be discussed in Chapter 4.
4. DESIGN METHODOLOGY

Major state-of-the-art design methodologies for global enterprise information system have been reviewed in Chapter 2. Based on the results specified in literature reviews, in this chapter, a new design methodology for global enterprise information system is proposed.

In the literature reviews, Malkoun and Kendall (1997) proposed the Cooperative Layered Agents for Integrating Manufacturing (CLAIM) Systems, which is a methodology for developing agent-based systems for enterprise integration, based upon the IDEF (ICAM Definition) approach for workflow modeling and analysis, the CIMOSA (Computer Integrated Manufacturing Open System Architecture) enterprise modeling framework, and the use of case driven approach to object-oriented software engineering. Furthermore, Brazier et al. (1998) presented a compositional development methodology (DESIRE) for developing object-based systems with features including a generic object model, graphical design tools, and mechanisms for dynamic object creation, which leads to our development of Hierarchical Design Pyramid (HDP). Therefore, the three steps approach – HDP, which consists of three components, namely business process reengineering, structural model design, and object-oriented design and analysis, is presented in this chapter.

4.1 Hierarchical Design Pyramid (HDP) – Three Steps Approach

Several methodologies are available for the design and modeling of an ERP system. Zhang and Alting (1991) suggested that the approach through structural modeling or
system approach is the first step towards the application of IT in manufacturing. Hargrove (1995) applied the structural approach to the design and planning of machine fixtures. Cheng and Robert (1996) used an IDEF model to develop a knowledge-based manufacturing information system. They have demonstrated a successful example for the system modeling using a structure methodology. In general, structural methods such as SSADM (Structured System Analysis and Design Methodology) and IDEF (Integrated Definition Methodology) are inadequate for problem domain analysis with natural concurrence. Therefore, some researchers (Coad, 1991; Jacobson, 1992; Martin, 1992) propose object-oriented methodology (OOM) to analyze and design the software systems to supplement the conventional structural methods.

Object-oriented methodology is considered to be an ideal approach to use in the construction of a large-scale system. Object-oriented design techniques and programming stress modularity and data encapsulation through narrow and rigidly defined interfaces as a way of achieving low coupling between individual software components. Coupling measures the interdependencies between components. Low coupling is a desirable feature for ERP systems because it increases the degree to which components can be modified, updated or replaced without affecting users of the component. Unfortunately, object-oriented designs are usually only achievable and enforced in a longer design period. This severely limits the success of the ERP system implementation.

In view of the limitation, the author proposes a Hierarchical Design Pyramid (HDP) model to resolve the limitations of both structural and object-oriented methodologies.
The basis of this model is to develop a paradigm that provides a culture supportive of change through IT enablement, quality management and internal and external communication networking. It also facilitates the implementation of effective process and change management practice throughout the entire company. In this model, the structural and object-oriented approaches have been integrated; these should be complemented with each other through a step-by-step systematic integration pattern. The advantage of this systematic approach is to ensure the design and implementation of a successful ERP system. The generic HDP model, which is composed of two
major building blocks – IDEF0, and Object-Oriented Modeling is used to achieve the ERP model (Ng, Ip and Lee, 1998). OOM can be easily used to maintain the structure and behaviour of the objects of the IDEF design that is derived from the ERP model. Figure 4.1 illustrates the model for the integration of technology aspects of ERP under the BPR context.

The HDP concept aims to adopt the OOM by capturing all the necessary states, methods, and the relationships between classes from the IDEF0 structural design. During the design phase of IDEF0, some elemental information such as input, output, control, mechanism, process and specification are extracted from the ERP model. This information is essential to the confinement of object models by reviewing the states, methods and the relationships among classes. Therefore, the IDEF0 design can be treated as the conceptual layer while the OOM is the physical layer.

The information captured in the IDEF function model can aid the system designer in the early stages of the development, particularly during requirements analysis. IDEF0 provides valuable insights into initial classes and routines that the system requires. It can also assist the system designer in scoping the system to see what should be included in the enterprise information system. This gives the development a more tangible start and finish point so that the system transitions from the development phase to actual implementation. IDEF0 helps the system designer understand the business of the end user, giving him information on the business. This is essential when making decisions on execution.
As IDEF0 modeling generally involves extensive use of function decomposition, the depth of decomposition is different from module to module. Normally, the decomposition should be two or three levels. Functions from the model represent methods. The concepts will be examined as classes or object instances. Reading the definitions and text included with the model also give the system designer more insight into what information will be needed in the system.

![Diagram](image)

Figure 4.2: IDEF0 methodology

In the diagram shown in Figure 4.2, the Create, Reference, Update, and Delete activities may correspond to Create, Reference, Update, and Delete methods in the OOD. If this particular diagram is decomposed any further, then the decomposition would be by type. It is possible that the Object concept may consist of a bundle of different kinds of objects which would be useful for defining the inheritance lattice in an OOD. The Authorization concept with subtypes Read Authorization and Write
Authorization could also be used in the OOD inheritance lattice. The Object input to Update and the Updated Object output is indicative of a state change and could be used in an OOD state model. The concepts Pointer, Object, User, Command, and Authorization could also be objects in the resulting OOD. Figure 4.2 illustrates how useful an IDEF0 model could be in an OOD. These models are also useful when used to validate designs and verify implementations, because they define what activities the system must support.

To sum up, the HDP methodology can be summarized as a three-step method (Figure 4.3) to design an enterprise information system for global manufacturing:

Step 1 (conceptualization): To perform a Business Process Reengineering to establish core requirements.

Step 2 (analysis): To execute structural analysis and design to develop a model of the desired behavior.

Step 3 (design and evolution): To construct the object design by using the results of analytical stage to create architecture and evolve the actual implementation.

![Diagram](image)

Figure 4.3: Three-step approach to Hierarchical Design Pyramid
In the following sections, the importance of BPR in the implementation of ERP and the design and implementation of the system by the proposed methodology will be explored.

4.2 Step 1: Business Process Reengineering

Most manufacturers today are so used to working according to conventional methods that they rarely ask themselves why they have to perform tasks in such ways. As a result, they are becoming less and less competitive when dealing with unpredictable changes. This situation is worsening since the business world is becoming complicated and globalized. Enterprises need to adopt radical improvement schemes in order to sustain their competitiveness in this global manufacturing era. Consequently, Business Process Reengineering (BPR) has become an extremely popular topic nowadays (David and Henry, 1995).

The idea of BPR first sprung from Michael Hammer in his article in 1990, 'Reengineering Work: Don't Automate, Obliterate'. Manganelli and Klein (1994) define BPR as the rapid and radical redesign of strategic, value-added business processes, and the systems, policies, and organizational structures that support them so as to optimize the workflows and productivity in an organization. They offer a definition that focuses on optimizing workflow and productivity in an organization. Hammer and Champy (1993) define reengineering as 'the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service, and speed'.
Many BPR practitioners indicated that the application of Information Technology (IT) is critical to the success of BPR (Tapscott and Caston, 1993; Kettinger et al., 1995). Hammer and Champy (1993) stated that, ‘A company that cannot change the way it thinks about information technology cannot reengineer’. They described the implementation of state-of-the-art IT as an essential enabler toward BPR. Manganelli and Klein (1994) noted that the appropriate methodologies of reengineering should feature both the empowerment of human resources and the use of IT as the prime enablers of radical changes. Davenport (1993) explained that the IT process has impacts in terms of organization streamlining/simplification, capturing and distributing, coordination, monitoring, analysis and decision making, and parallelism enabling process change.

With the accelerated advancement of technology, a manufacturing technology that works to drive down production cost and an IT that enables massive amounts of data to be transmitted and processed at an unprecedented rate, have changed many of the conventional ways of doing business. Tapscott and Caston (1993) describe three fundamental shifts for an organization in the application of computers in business, each affecting a different level of business opportunity. They state that IT enables enterprises to evolve from a ‘high-performance team structure’, to function as an ‘integrated business’, then to develop further new relationships with external organizations, becoming an ‘extended enterprise.’ Venkatraman (1994) proposes a similar framework that breaks IT-enabled business transformation into five levels.

The level of potential benefits achieved by adopting different IT strategies is proportional to the degree of business transformation undertaken. According to
various literatures (Brown and Vessey, 1999; Koh, Soh, and Markus, 2000), the use of IT in BPR has promoted a successful rate of implementation. However, the literature very often addressed the significance of IT in reengineering but a pertinent model is required to design and implement an ERP system under a BPR context. In contrast, an implementation paradigm for ERP and BPR integration with four different dimensions was proposed, namely Fundamental, Process, Radical, and Dramatic. The author identified that embedded BPR should allow the setup of workflow according to the organization processes and users requirements. In general, in order to prepare the design and implementation of the proposed enterprise information system for global manufacturing, the seven steps should be followed:

(i) Describe the project (establish boundaries)

(ii) Create vision, values and objectives

(iii) Redesign business processes and tools (model)

(iv) Evaluate concept (benefits statement)

(v) Plan for implementing the solution

(vi) Implement the redesign

(vii) Transition to continuous process improvement (measure results)

Through the execution of BPR, enterprises will find that the proposed model is incorporated with the concept of BPR to support ERP that includes four key elements,
namely fundamental, process, radical, and dramatic. These four elements are summarized in the following sections.

4.2.1 Fundamental

Fundamental implies that everything which the organization has been doing — every assumption, every reason, every process and every activity — is challenged by asking why it should be done in such a way. This concept enables the refinement of system specifications by continuously asking ‘why’. The set of system specifications is primarily important to the accuracy and consistency of the business rules and logic in the ERP model.

4.2.2 Process

Process is defined as a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer. Due to the success of the division of labour in an organization, people become more concentrated on the tasks performed around them. Then, in the system development stage, the personnel can focus on their own tasks by using the structural methodology (IDEF0). The components in IDEF0 such as input, output, control and mechanism can be defined precisely.

4.2.3 Radical

The major difference between continuous improvement and reengineering is that the latter is not about improvement at all, but about reinvention. The idea of reinvention is crucial to the Object-Oriented paradigm. The conventional way of doing something
may not be the most appropriate way. In OOM, the object models and the relationships between them can be elaborated upon in the initial stage.

4.2.4 Dramatic

The intention of reengineering is to achieve quantum leaps in performance. It is not about making marginal or incremental improvements, the achievement of which can be brought about by other more conventional methods. The ERP system can provide manufacturers with a significant improvement in terms of inventory level, unit production cost, quality, etc. The HDP model also contributes to an effective way for system development and implementation by the integration of structural and object-oriented approaches.

The following section describes a conceptual model that not only encompasses the management concerns of BPR and continuous improvement but also the technology aspects of Object-Orientated models and Internet-enabled applications. It provides an essential guideline for the design, development and implementation of ERP whether it is large, medium or small business organizations.

4.3 Step 2: Structural Model Design

In the HDP model, a complete IDEF0 model of the ERP system is produced in an interactive process by determining the yet-to-be modeled boundaries of the system. The main entities and data that pass to and from the system must be identified. The first draft of a top-level A0 diagram therefore can be produced. It is important to remember that the IDEF0 model may only be developed upon one defined perspective.
If the focus is lost, the quality of the model is reduced. From this defined perspective, the 'view' is decomposed and a consistent decomposition can be developed. Furthermore, the purpose of the model must be clarified on the A0 diagram. The level of detail depends on the defined purpose. This is thus the criterion for determining when to stop decomposing the model. The drafted diagrams can be circulated around the relevant departments, where each department checks and judges how accurately their parts of the system have been modeled. The input received from individual departments act as feedback information so that the system designer can make amendments. The cycle is then repeated several times until the model is completed and approved. Figure 4.4 illustrates the general model representation.

![Diagram](image)

Figure 4.4: IDEF0 general model representation.

4.3.1 IDEF design

Because ERP links the business strategies, organization structure, business processes and IT together in an integration framework, the HDP provides a systematic approach to implement these elements. By following HDP, the management and technologist are able to use the existing tools and techniques for the customization of the ERP. It enables the development of an analytical framework for the mapping of customization
parameters of the ERP “AS IS” model to the “TO BE” models. On the other hand, software developers can provide a quick customization method based on the latest management and technology in a case-by-case scenario.

Figure 4.5: Node tree of the generic ERP model from the HDP approach.
Figure 4.5 depicts the hierarchical structure of the Generic ERP system derived from the proposed HDP model. This is generated by Design/IDEF software and illustrates the node tree decomposition of the model, giving the relevant titles and page numbers.

IDEF0 is used to establish the scope of a functional analysis of the ERP system. The tool also enhances domain expert involvement and consensus decision-making through simplified graphical devices. It assists the modeler in identifying what functions are performed, what is needed to perform those functions, what the current system does right, and what the current system does wrong. Accordingly, all the activities involved in the design of an ERP system are modeled by first defining the most important Inputs, Outputs, Control functions and required Mechanisms (IOCM).

Figure 4.6 shows the top-level diagram, A0 of the model.

![Diagram of A0 building a generic ERP model](image)

Figure 4.6: Node A0 building a generic ERP model.
(A) Construction of the system planning model

With reference to Figure 4.7, the project begins with system planning. This is a blueprint that organizes the resources and directs future activities involved in the project. The three activities consist of Project Scope and Goal Definition, Feasibility Study, and Final Project Plan. The details can be elaborated as follows:

![Diagram](image)

Figure 4.7: The decomposed page from the block A0 of the generic ERP model.

1) Project scope and goal definition

An initial study of the business operations is conducted in order to arrive at a problem definition. The project group then is able to define the scope of the project and settle for a certain set of goals against which the alternative design is evaluated. The operating requirements, data requirements and the processing requirement is also obtained as input for the feasibility study.
(2) **Feasibility study**

The constraints and controls of the project are located. These are carefully examined in order to decide whether the preliminary requirement defined at the earlier stage can be fulfilled.

(3) **Final project plan**

The plan specifies the scope, system performance, timing and budget allocated to the project. A timetable is produced to organize and control the resources and the activities.

(B) **Process reengineering**

This stage of process reengineering calls for a study and redesign of the existing state of affairs in a given business area of the enterprise. It begins with preliminary analysis and the business operations and problems are studied in greater detail. The process reengineering includes four steps that are elaborated upon as follows:

(1) **System analysis**

The system designers must completely familiarize themselves with the current working system by auditing its flexibility, reliability, accuracy, efficiency and connectivity. Any problems that are inherent in the system can be fully understood and clearly defined. Modeling techniques are employed to map out the logical flow of the processes involved in the system.
(2) **Process re-thinking**

This involves recognizing potential solutions, and then seeking and recognizing obvious or latent problems that may be solved. The system designers should query the assumptions that underlie current business operations in order to generate new methods. In this way, a set of new system rules and processes can be achieved for process benchmarking.

(3) **Process benchmarking**

Benchmarking is usually considered a continuous improvement tool (Büyüközkan, 1996). In this model, it can be used to gain information regarding a company's relative position in key business processes and core competencies. The project group can use this technique to create an industry context for the goals being set. This provides the company with examples of best practices in terms of new processes and the approach to implementation.

(4) **Simulation**

Simulations are used to visualize and evaluate the redesigned processes generated by process re-thinking. This provides reengineers with an appropriate tool for evaluating new processes. Discrete-event computer simulation and animation can be accurately applied in understanding processes. After the evaluation, a new system description including user requirements and specifications and defining what needs to be done, given the support of more advanced information systems, is produced and is used in the system design stage.
4.3.2 Quality Walk Through Procedure

A Walk-Through Procedure has been proposed as a guide for presenting model information to a group of "reviewers" in order to ensure the quality. It does not substitute for the Reader/Author Cycle review process that is central to the quality assurance of IDEF0 modeling, but will be streamlined for periodic project use at the technical level to provide an opportunity for all participants to share or develop common interpretations. The procedures of quality walk through are listed as follow:

(i) Present the model to be analyzed by using its node index. This is the model's table of contents. Provides a quick overview of what is to come.

(ii) Present selected glossary terms. Encourage each reviewer to replace personal meanings of words with those that the presenting team has chosen. The meanings should not be questioned at this point. A change in meaning now would require many changes in the diagrams.

(iii) Present each diagram for review.

The diagram walk-through process is an orderly, step-by-step process where questions can be asked that may identify potential weaknesses in the diagram or its text. Diagram corrections may be proposed at any step. These corrections may be noted for execution at a later date or adopted immediately. Four stages of a structured quality walk-through are described below:
(A) Stage 1: scan the diagram

This stage allows the reader to obtain general impressions about the content of the diagram. Typically, the reader will have reviewed the parent diagram which depicted the current diagram as one of its boxes. The reader is now examining how the author decomposed that function according to the following criteria for acceptance:

(i) The decomposition is useful for its purpose and is complete within the context of its parent box. All lower level functions can clearly be categorized under each of its boxes.

(ii) The diagram reflects, in the reviewer's opinion, a relevant point of view based on the purpose of the model.

(iii) In the opinion of the reviewer, there is enough new information provided to extend understanding of the parent box. There is not so much detail that the diagram appears complex and hard to read.

Unless a problem is rather obvious, criticism may be delayed until Stage 3 below. However, first impressions should not be lost. They might be put on a blackboard or flip chart pad until resolved.

(B) Stage 2: look at the parent

Once the reader understands the current diagram's decomposition, the parent diagram should be reviewed to insure compatibility. It must ensure that the decomposition covers all of the points the reviewer anticipated when reading the parent diagram.
Now that the decomposition of this portion of the parent diagram is revealed, the detail which the reviewer envisioned for this box should still seem correct. If not, note the missing detail.

It might be important at this stage to return to the parent diagram briefly and add new notes or embellish existing ones, based upon the added insight gained from this look at the decomposition.

(C) Stage 3: connect the parent box and the detail diagram

This stage tests the arrow interface connections from the parent to child. Reviewers have to ensure that:

(i) There are no missing or extra interface arrows.

(ii) Boundary arrows are labeled with proper ICOM codes.

(iii) Child arrow labels are the same or an elaboration of its parent’s matching arrow. Labels convey the correct and complete arrow contents.

(iv) Examination of the connecting arrows reveals no problems in the parent diagram.

(An added interface may create a misunderstanding of the message conveyed by the parent.)
A clockwise tour of the four edges of the parent box, checking each arrow, will provide a methodical way to check matching of ICOM codes boundary arrows to the parent arrows.

(D) Stage 4: examine internal arrow pattern

The pattern of boxes and arrows constitutes the primary expression of the model being created. Each box will be examined in node number order, and each arrow followed in ICOM order for each box. When this process is complete, the reviewers should be led through the diagram to explore the consequences of situations with which reviewers are familiar and to test the diagram’s capability to simulate the relationships known to exist. The criteria for acceptance are listed as follows:

(i) The diagram does not look cluttered. The number of arrow crossings and bends is minimized.

(ii) The boxes should be balanced with regard to detail. There should be an equal amount of detail within each box. However, compromises on this criterion are acceptable for the sake of clarity.

(iii) The diagram should be consistent with the reviewer’s experience and knowledge of the subject matter. Feedback and error conditions should be shown as the reviewer expects.

(iv) The level of detail of the arrows should match the level of detail of the boxes. Bundling of arrows into more general arrows should be considered.
4.4 Step 3: Object-Oriented Design And Analysis

With reference to the proposed HDP (Figure 4.1), the third step is to undergo an object-oriented design and analysis. This step collects the input, output, control, mechanism, process, and specification from the IDEF0 functional design of Step 2 in HDP so as to transform the object model based upon the design specification of HDP.

4.4.1 Object model

The proposed methodology consists of two distinct models, namely hierarchical static model and ladder dynamic model. The class-diagram includes all necessary static information of an object model, which represents the static model. The object-diagram shows how the classes interact with each other; it captures some moments in the life of the ERP system and helps to describe the dynamic behavior.

In relation to the structural model design, the knowledge captured in the input and output of the IDEF0 can be used to construct the static model. The dynamic model is formed by the information extracted in the control and mechanism of the structural model. The proposed HDP model supports the iterative and incremental development of an ERP system. In order to develop the entire object model in different modules of the ERP system, the following procedure has been identified:

(i) Identify the classes and objects at a given level of abstraction

(ii) Identify the semantics of these classes and objects.
(iii) Identify the relationships among these classes and objects.

(iv) Specify the dynamic behaviour of these classes and objects.

(v) Specify the interface and then the implementation of these classes and objects.

(A) Hierarchical static model

The hierarchical static model specifies the hierarchical structure of the static design objects (i.e. the design schema). It includes information such as the objects that will be needed in the design and instances of those objects, attributes describing the objects, classification of objects into sub- or super-classes, and hierarchical relationships and roles among the objects. Examples of the information depicted in a static model could include the following items:

(i) Subclass/superclass relationship of class objects; e.g., an employee is a subclass of the superclass person,

(ii) Operations that can be performed on an object; e.g., a document can be printed, a person can be hired, an employee can be assigned,

(iii) Two way relationships; e.g., employs/employed-by relationship between class objects employee and employer, and

(iv) Attributes of the class object; e.g., the class object person may have attributes of age, name, weight, and height.
(v) The model includes object specifications and hierarchical connections between objects. These connections include inheritance, relationships, and links.

(B) Ladder dynamic model

The ladder dynamic model specifies communication between objects and transitions between states in a ladder-like format. It contains a graphical depiction of the messages that are relayed between objects, events which cause the object to implement the message, and the resulting transition of the object from one state to another as the object executes the message. Examples of communications that can be specified include the following items:

(i) An object instance "switch" transitioning between the states "on" and "off."

(ii) The "open" event generated by clicking on a file icon and the "launch" message sent to its corresponding application.

(iii) The "door-open" event generated by a sensor software component and the "sound-alarm" message sent to the security management component, and

(iv) A conductor sending a "play louder" message to members of the orchestra.

4.4.2 Formulation of hierarchical static model

The structural design of an inventory management sub-system is elaborated upon in the Section 4.3, which includes the conceptual design and behavioral analysis. In order to create system architecture, according to HDP, an object-oriented approach is
used. Figure 4.8 shows the class relationship diagram of the item master module in the inventory management sub-system, which is used to illustrate the application of step 3 in HDP approach.

![Class relationship diagram of item master module](image)

Figure 4.8: Class relationship diagram of item master module

The basis of this class diagram is *Item* which is identified as an Abstract Base Class (ABC). The super-class *Item* inherits three sub-classes namely *Manufacturing Item*, *Trading Goods*, and *Fixed Asset*. They all follow the “IS_A” relationship as mentioned by Booch (1994). The main idea demonstrated here is the characteristic of inheritance. The diagram depicts the fact that the sub-class *Manufacturing Item* inherits another five sub-classes, but they also inherit the states and methods from the very super-class *Item*. The coding procedures can then be implemented easily by this object-oriented technique.
The other important characteristic of the object-oriented approach is polymorphism.

Figure 4.9 describes a part of the class relationships of the Purchasing/Receiving System, which has been discussed in the previous section.

Figure 4.9: Class relationship diagram of sales and purchasing system.

Four individual sub-classes namely *Purchase Order, Sales Order, Delivery Note*, and *Receive Note* are inherited from the Abstract Base Class Document Template. The sub-classes always perform the same operations, for example, print, add, edit, etc.
Polymorphism provides a generic mechanism to perform various operations on different objects. In this illustration, when operation print() is performed on an object sale order, the actual operation is determined by the type of object the operation addresses. Then, the programmer only needs to implement the actual coding into the object itself, so that a higher degree of abstraction in system development is allowed. It also separates the design of object operations from the actual implementation. In C++ programming language (Stroustrup. 2000), the VIRTUAL function is used to implement the polymorphism. In other Object-Oriented Programming (OOP) language such as Ada, this fully supports polymorphism throughout the whole system. In this research project, the author proposed a new methodology – WOOM to implement the object model which will be elaborated in the next chapter.

4.4.3 Formulation of ladder dynamic model

The preceding section discusses the design issues involved in the purchasing/receipt system. In this section, an object diagram is used to describe the dynamic behavior of the purchasing sub-system. An object diagram representing various scenarios of purchasing an item from a vendor is described in Figure 4.9. This is the simplified object diagram that introduces the main idea of design by scenarios. The scenarios start by notifying the buyer of the need to order such items as parts or materials. The buyer then checks the history of vendors’ performance and selects the best vendor for such materials. He contacts the selected vendor so as to confirm the availability and the cost of these materials. Following the scenarios, Figure 4.10 can be constructed. This method can help the ERP analyst to understand the detailed programming functions of the system and identify all the attributes and operations concerned in all of the classes.
Figure 4.10: Dynamic object diagram of purchasing an item

By thinking of scenarios that are in actual use by the ERP system, the context for the object diagrams and interaction diagrams can be generated. These scenarios can help to find out the required operations in the classes, or verify their correctness and sufficiency. The class diagrams or the overall design can then be refined recursively. Figure 4.11 shows the Interaction Diagram of the Object Diagram of Figure 4.10. This diagram is used to refine the design and implementation of the dynamic interaction of the objects.

Figure 4.11: Ladder diagram of purchasing an item
The HDP method is designed to assist in the correct application of object-oriented technology. The most important reason for deploying the HDP method is that it views object-oriented design as part of a larger system development framework, rather than an object-oriented analysis and design method that is everything to everyone. It stresses the object-oriented design procedure over the graphical syntax, using the graphical syntax and diagrams as aids to focus on and communicate important design issues.

4.4.4 Inheritance

This section further elaborates on the inheritance theory as applied to the static model stated in Section 4.4.2. According to the proposed design methodology, an object hierarchy is formed after the formation of the hierarchical static model. A class is a set of objects that share a common structure and a common behavior that is the same type of equipment in the object hierarchy. Inheritance is used to reduce the complexity of the object hierarchy by eliminating duplicate and similar objects. There are many different relationships possible between classes. The most important of all these relationships is inheritance. Inheritance is appropriate between classes only when we can have "IS_A" relationship between the objects. By the inheritance law, if a subclass (or derived class) inherits from the superclass (or base class) then an object of the subclass is an object of the superclass, but not vice versa. In single inheritance, one subclass inherits from one superclass. In multiple inheritances (see Figure 4.12), one subclass inherits from more than one superclass. No matter in single or multiple inheritances, inheritance relationships cannot form a cycle.
As a design strategy, inheritance introduces an intuitive design. Objects constitute the fundamental building block for modeling. After the object hierarchy is formed, the inheritance relationships between each node could be identified. There are three major types of class relationships in an object hierarchy.

(i) Parent-to-Child Association (e.g. Purchase Order contains Item). The most common relationship between objects is that of association (or aggregation). In association relationship, objects relate to one another in a whole part hierarchy. That is, one object contains another object.

(ii) Parent-to-Child Inheritance (e.g. employee is a person). The parent-to-child inheritance relationship exists when the child node is inherited from the parent node. In this case, the parent node is described as an abstract base class and it is a pure logical node in the object hierarchy.
(iii) Node-to-Node Inheritance Hierarchy (e.g. vendor A and vendor B are both inherit from the vendor class). The node-to-node inheritance hierarchy is a kind of inheritance as well. However, the abstract base class does not exist in the object hierarchy. A separate n-way 3D graph is required to build on top of the existing object hierarchy.

Inheritance fully facilitates re-use of code. Thereby minimizing total coding effort needed. Especially in an ERP system, each type of component usually has more than one piece and the layout for each subsystem is expected to be similar, inheritance reduces the total implementation effort. It can be fully demonstrated in the next chapter.

4.4.5 Polymorphism

Another special feature of the proposed design paradigm is its Internet-based development capability, which is essential to be adopted in the global manufacturing environment. This section further elaborates on the polymorphism theory as applied to the hierarchical static model. Polymorphism is a Latin word that means many (poly-) objects of such form (~morphism). Basically, polymorphism is a concept in type theory such that a name may denote instances of many different classes as long as they are related by some common superclass. Any object denoted by this name is therefore able to respond to some common set of operations (Booch, 1994). In short, different objects could be treated in the same way by the law of polymorphism. The nature of polymorphism is employed as the open system (Nicoloro, 1994) architecture for handling different forms of implementation such as design methodology, programming language, performance requirement and operating system.
In the presence of polymorphism, new design and implementation strategies could be employed. The following are four derived techniques that are based on the law of polymorphism.

(i) Design polymorphism

(ii) Implementation polymorphism

(iii) Performance polymorphism

(iv) Browser polymorphism

Design polymorphism is the most common technique to be employed in OOAD. This kind of polymorphism separates the detail implementation level from the abstract design level. It also facilities system designers to combine a mountain of similar operations through a simple handling procedure. For instance, the operation to turn the light on and the operation to turn a heater on are both described as a “turn on” operation in the point of view of a system designer. Therefore the total number of different operations could be highly reduced (Gay, 1994).

Followed by the design stage, implementation polymorphism is the actual phase to built polymorphism by means of coding. This type of polymorphism is language dependent. For example, in C++, virtual function is the feature to support polymorphism as mentioned in Section 4.4.2. By declaring a base class to be virtual, it implies that classes derived from the base class may have their own version of
operations in the base class and these operations are invoked based on the actual object types at run-time.

Performance polymorphism refers to the concept of maintaining and selecting appropriate implementation approach. The selection criteria are usually constrained by execution time, memory space, system configuration, result precision, and so on. In a real-time system, this type of polymorphism is essential if timing constraint is important.

For enabling the feature of Internet-based development, the technique of browser polymorphism is employed. This type of polymorphism is usually integrated with the web browser. With browser polymorphism, the code generation operation could then generate appropriate structural markup language (SML) according to the specified browser. The most important aspect in here is that adding new browser generating operation will not affect the existing browsers.

In short, a new design methodology – HDP – has been proposed and introduced in this chapter. The next chapter will focus on the implementation approach.
5. IMPLEMENTATION APPROACH

As described in the previous chapters, making effective use of information technology has become critical for companies to lower the cost of doing business and of obtaining competitive advantages in a global manufacturing environment. The software industry was drowning in complexity and failing to deliver many projects on time or to budget. Furthermore, organizations were finding ERP difficult and costly to support and make enhancements. Such code was frequently written in a monolithic fashion such that completely different bits of logic were intertwined and this made it risky to make changes — even what might appear to be a minor amendment could easily unknowingly cause severe repercussions to ripple throughout the system.

5.1 Web-Based Object-Oriented Model (WOOM)

In the HDP model (see Figure 4.1), ERP implementation is realized through Object-Orientated (OO) approach, which is an evolving paradigm in the computer industry. The object-oriented method represents a model of a system that is based on real world entities. Objects can represent organizations, incidents, or roles which individuals or organizations play (Sherif et al, 1994). Object-Oriented Technology (OOT) also employs the principle of inheriting characteristics or attributes from super class objects. The inheritance mechanism of OOT supports reusability of software, and simplifies design. Many literatures (Stienen and Van der Weerd, 1989; Ji, 1991) mentioned the employment of object-oriented approach to model and develop manufacturing systems. In this chapter, the implementation of HDP-OO model with
Web-based technology is further elaborated and is called Web-based Object-Oriented Model (WOOM).

WOOM is a development process that emulates the construction process used by many other established industries. To cite an example, the majority of today’s automobile manufacturers build vehicles from hundreds of component parts supplied from a vast number of external vendors. Each individual part would have been designed, manufactured and passed through quality checks as a separate isolated process undertaken by the part vendor. By removing many low level complexities from the car manufacturer, it enables them to concentrate on their core business – that is, designing and building cars that are more reliable and can be brought to market more quickly, thanks to a shorter development cycle with the highest quality. The proposed implementation approach is to build a large ERP system in a similar manner – that is by building a number of items of software, each of which performs a specific item of functionality and gluing them together to form a single integrated solution through the proposed HDP model.

5.2 Model Overview

A WOOM component can be defined as a combination of data and code, which acts off that data, and together, can be considered as a single unit. The data and code associated to the component defines everything that the component represents (state) and what is can do (behavior). In general, WOOM provides the framework, channel and the interface necessary to build a Web-based ERP system on a global business environment. By adopting advanced development techniques, a sophisticated, but easy to use building model that provides a total solution for all areas of a global ERP
system can be developed. The model is formed by four major object domains. The proposed architecture adopts the object technology, which aims to enhance the scalability and configurability of the system. The WOOM’s characteristics can be represented by the terms:

(i) Properties – these are the attributes or the object and represent the data that is encapsulated within it; sometimes properties can be accessed or changed from outside the object.

(ii) Methods – these are functions that can be performed internally, usually upon the encapsulated data; sometimes methods will take arguments.

(iii) Events – these are signals fired when a specific action or interesting state occurs; these signals are detected outside of the object.

With the above characteristics, the architecture can be logically divided into four major domains, (a) Logic Domain (LD), (b) Presentation Domain (PD), (c) Service Domain (SD), and (d) Database Domain (DD). The object domains of the architecture are characterized by their high degree of flexibility taking advantage of the object technology. The high level of integration of the individual applications assures security and consistency of data throughout the Internet. WOOM objects are the building blocks of ERP. Each WOOM object will carry out a unique and complete operational function. For example, an inventory management system is a single WOOM object. Operations and essential data are all defined within that inventory object. Another WOOM object such as an accounting object can
interchange information with the inventory object through the interface provided by the SD. The architecture is depicted in Figure 5.1.

![Diagram of WOOM architecture](image)

**Figure 5.1:** A single WOOM object.

The model is divided into two distinct layers. They are Real Object layer that defines the user's conceptual view and Application layer that defines the corporate resources view. The Real Object layer belongs to the Presentation Domain while the Application layer exists in the Logic Domain. The user's conceptual model of a WOOM object is not a direct reflection of the Application model. The Application model is represented as an IDEF Model mentioned in the previous chapter. The IDEF model is a master blueprint defining all of the business entities, relationships, and attributes that must be maintained to preserve the integrity of the corporate information resources. The internal operation of WOOM objects is depicted in Figure 5.2.
Figure 5.2: Interaction between WOOM objects.

The WOOM object is closer to the user's concept of the data representation than the IDEF Model, and showing only the portion of the master blueprint that is relevant to
the business activity in which the user is engaged. The conceptual model of a WOOM object takes the shape of a hierarchical composite object that is identified by the root subject entity. For example, the Purchase Order is a form to fill out which has some header information and line items but the IDEF Model will define a Purchase Order as the combination of multiple entity types with connections to many related entities.

Every function in the ERP system can be modeled by a single WOOM object. In Figure 5.2, three WOOM objects are illustrated, including inventory object, sales object and purchasing object. The internal communication of these three objects is by the means of message passing through the Service Domain. The most up-to-date enterprise information, such as inventory level, is shown on the user’s web browser by the Presentation Domain. In order to retrieve the real-time data, Logic Domain requests the Database Domain to extract all the necessary data from the corporate database engine by the database query generated from the Logic Domain. The function and features of the model components are elaborated in the following sections. The “lock” symbols in the diagram indicate that all the data passing through the database server and web browsers are being secured by encryption.

Basically, the Presentation Domain is concentrated on the interaction with the user and managing the user’s interface. The Logic Domain is focused on managing the corporate resources (e.g. database tables) that users share, in particular, the application logic. Only the Logic Domain can directly interact with the Database Domain. The Database Domain is a generic interface to the relational database server (e.g. Oracle, Informix, SQL Server, etc.) that manages the persistent storage of data.
This domain is also responsible for preparing of any report, graph, or query. The following sections will further elaborate on the key elements of the proposed system architecture.

5.3 Logic Domain (LD)

The Logic Domain (LD) is the central processing unit of the model, which has methods and properties that offer general utility functions. The implementation of LD represents the environment in which the application runs, and the other domains are used to make up an ERP application. Major functions of this domain include:

(i) Generating presentation mimics

(ii) Providing program logic

(iii) Communicating and Synchronizing with databases

(iv) Processing service requests from the Service Agent

(v) Routing global requests to the Request Agent

(vi) Managing concurrent users

The application components of Logic Domain are Entities, Relationships, Elements, Transactions, and Databases (Ng, Ip and Lee, 1997). The architecture is presented as Figure 5.3.
Figure 5.3: Logic domain.

It also acts as the interchange between the Web server on the open network and the Database server on the private network. It is used to manage the corporate database, maintain its integrity in order to maximize the usefulness of it, and provide program logic to cater for the business operation. This domain is designed to retrieve data in a variety of useful ways and update the database in a safe and controlled manner. It also ensures that the event-driven processes are triggered and the users obtain up-to-date data in their workstations.

The implementation of IDEF Model is the backbone of the Logic Domain architecture. The semantics, relationships, and the dynamic behaviors of the classes and objects are specified through the IDEF design in HDP model. The Entities object and Relationships object receive a message, process it, and return a reply. Moreover, the architecture can store information that many scripts need to access globally and can store objects that many scripts will access. The hierarchical object model is shown as Figure 5.4.
Figure 5.4: Hierarchical object model.

Relationship in Logic Domain is also responsible for spreading the event driven business process. It also provides a query method that will produce a list of the members' Relationship. The query specifies which column to include, ordering, criteria and volume specification. Moreover, the Relationship can assign the sequential key to Entity by query the database. The Relationship is also used to resolve concurrency locks and register interest.

The Logic Domain enables the creation of any external objects the web server needs to do its work. It can set the timeout value so that long tasks can be completed without interruption. Operations automatically timeout in a specific time frame if they do not complete their task to conserve server resources being consumed by runaway scripts, infinite loops and the like. Figure 5.5 depicts the constitution of logic domain.
<table>
<thead>
<tr>
<th>Constitutes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScriptTimeout</td>
<td>The amount of time that a script can run before it times out.</td>
</tr>
<tr>
<td>LCID</td>
<td>The locale identifier.</td>
</tr>
<tr>
<td>SessionID</td>
<td>Returns the session identification for this user.</td>
</tr>
<tr>
<td>Timeout</td>
<td>The timeout period for the session state for this application, in minutes.</td>
</tr>
<tr>
<td>CreateObject</td>
<td>Creates an instance of a server component.</td>
</tr>
<tr>
<td>SMLEncode</td>
<td>Applies SML encoding to the specified string.</td>
</tr>
<tr>
<td>MapPath</td>
<td>Maps the specified virtual path, either the absolute path on the current server or the path relative to the current page, into a physical path.</td>
</tr>
<tr>
<td>URLEncode</td>
<td>Applies URL encoding rules, including escape characters, to the string.</td>
</tr>
<tr>
<td>Abandon</td>
<td>This method destroys a Session object and releases its resources.</td>
</tr>
<tr>
<td>Session_Start</td>
<td>The Session_Start event occurs when the server creates a new session.</td>
</tr>
<tr>
<td>Session_End</td>
<td>The Session_End event occurs when a session is abandoned or times out.</td>
</tr>
</tbody>
</table>

Figure 5.5: Constitution of logic domain.
5.4 Presentation Domain (PD)

Presentation Domain is mainly concentrated on the manipulation of Web-based user’s interface and interaction with the user, i.e. the Web browser. The architecture is presented as Figure 5.6.

![Diagram](image)

Figure 5.6. Presentation domain.

Presentation Domain is the standard man-machine interface employed by the web browser. The concept of this domain is simple – PD uses a web browser as the client to access all applications. Texts, graphics, animations, multimedia effects, charts and dynamic data are all presented with the help of a web browser. This design is based on a thin-client architecture. This means that no application resides on the client; all application development is performed on the server. This architecture reduces the application maintenance effort and the total cost of ownership of an ERP system and allows worldwide clients to be connected to the system with no configuration effort.

In order to provide a standard interface, all graphical user interfaces are based on the structural markup language syntax and format. PD transforms texts that are structural
markup language tags, including HTML and XML, into the correct codes to display properly. In addition, it encodes structural markup language so that characters not allowed in a browser parameter can be used without incident. Figure 5.7 depicts the constitution of logic domain.

<table>
<thead>
<tr>
<th>Constitutes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CodePage</td>
<td>The codepage that will be used for symbol mapping.</td>
</tr>
<tr>
<td>Contents</td>
<td>Contains all of the items added to the display through script commands.</td>
</tr>
<tr>
<td>ContentType</td>
<td>HTTP content type (i.e. “Text/HTML”).</td>
</tr>
<tr>
<td>SMLEncode</td>
<td>Applies SML encoding to the specified string.</td>
</tr>
<tr>
<td>URLEncode</td>
<td>Applies URL encoding rules, including escape characters, to the string.</td>
</tr>
<tr>
<td>Start</td>
<td>Occurs when a presentation in the application is first displayed.</td>
</tr>
<tr>
<td>End</td>
<td>Occurs when the application ends.</td>
</tr>
</tbody>
</table>

Figure 5.7: Constitution of presentation domain.
5.5 Service Domain (SD)

Service Domain (SD) is consisted of two agents: they are the Supply Agent and the Demand Agent. The Supply Agent is an external communication interface that enables the Logic domain to release its public services for another WOOM object. Another object can request service from the Supply Agent of any object through its own Demand Agent. Generally speaking, the Supply Agent together with the Demand Agent provides a standard communication protocol for inter-object communication.

An important reason of having a Supply Agent is that it can help to objectize the model. The Supply Agent is the only channel for another object to make a request. Hence, it is easier for system integrators to develop their own services on top of the standard architecture. In general, the Supply Agent is responsible for passing message to another Demand Agent. It can transfer and halt the message being processed in midstream if required. The Supply Agent is also a buffering component as it can control whether a message is served to the Presentation Domain in one or many transfers.

The Demand Agent is an external communication interface that enables the WOOM objects to request information from global objects. Once a request is made, the Demand Agent will seek for the information through an intelligent routing mechanism. The Demand Agent can be considered as an intelligent router for information interchange. Intelligent agent is used to determine load balancing; heavy loading services can be downsizing into concurrent services on the Internet/Intranet. This approach can highly reduce the server hunting condition. It also provides manageability where any services are now treated as global objects. Hence, system
integrator can develop their own services without knowing the details and physical location of its depended services. Figure 5.8 depicts the constitutions of service domain.

![Service Domain Diagram]

<table>
<thead>
<tr>
<th>Constitutes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TotalBytes</td>
<td>Specifies the number of bytes the client object is sending in the body of this request.</td>
</tr>
<tr>
<td>Buffer</td>
<td>Indicates whether to buffer the signal until complete.</td>
</tr>
<tr>
<td>Status</td>
<td>Value of the request status returned by the LD.</td>
</tr>
<tr>
<td>IsConnected</td>
<td>Indicates whether the requesting object has connected to the communication channel.</td>
</tr>
<tr>
<td>BinaryRead</td>
<td>Used to retrieve data sent to the LD as part of the request.</td>
</tr>
<tr>
<td>Redirect</td>
<td>Instructs the object to connect to a different location.</td>
</tr>
<tr>
<td>AppendToLog</td>
<td>Adds text to the LD log entry for this request.</td>
</tr>
</tbody>
</table>

Figure 5.8: Constitution of service domain.

### 5.6 Database Domain (DD)

Database Domain (DD) is a standard interface for accessing the physical databases. It offers an excellent mechanism that provides access to data of all type through Open Database Connectivity (ODBC). It is the only interface in the system that is
responsible to retrieve, manipulate and update data directly to/from the database system through a Transaction object, which is the controller of all database updating activities. The function of a Transaction object is to ensure that the database is always recoverable to a logically consistent state. The transaction agent can be aborted at anytime and the updates rolled back when error signal is detected by any object. The operation of Transaction is represented as Figure 5.9.

![Diagram of Transaction Operation](image)

**Figure 5.9**: The logical flow of transactions.

DD is used to provide multiple and different database systems support. It isolates the Logic domain from the physical database and so different database systems can cooperate seamlessly and simultaneously. Moreover, DD acts as the only channel for database access which separates the physical database from the Internet connection such that the database is protected from illegal access and security is achieved.
One of the major advantages of DD is to allow scalability; the systems developed by system integrators who make use of the model can take the advantage of universal database support, ranging from personal file-mapping database to those high-end client/server database engines. This Domain is also responsible for preparing of any report, graph, or query. It is comprised of three components, namely, presenter, extractor, and analyzer. Presenter is used to present all the report, graph, or query to the user by all available media (e.g. screen, printer). Extractor can communicate with Logic Domain through a Database object by calling the Database Management System (DBMS) to perform all SQL operations. Analyzer assimilates the data and prepares it for presentation. All of the calculation and sorting operations for a report are performed by this component. Figure 5.10 depicts the constitutions of database domain.
<table>
<thead>
<tr>
<th>Constitutes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes</td>
<td>Controls whether to begin a new transaction when an existing one ends.</td>
</tr>
<tr>
<td>Connection</td>
<td>The information used to create a connection to a data source.</td>
</tr>
<tr>
<td>Mode</td>
<td>Sets or returns the database engine’s access permissions.</td>
</tr>
<tr>
<td>State</td>
<td>Returns whether the connection is open or closed.</td>
</tr>
<tr>
<td>Open</td>
<td>Opens a new connection to a data source.</td>
</tr>
<tr>
<td>Close</td>
<td>Closes an existing open connection.</td>
</tr>
<tr>
<td>Execute</td>
<td>Executes a query, SQL statement, or stored procedure.</td>
</tr>
</tbody>
</table>

Figure 5.10: Constitution of service domain.

In summary, this chapter presented an innovative implementation approach – WOOM, which includes four domains, namely Logic Domain, Presentation Domain, Service Domain, and Database Domain. Based on this implementation approach and design methodology developed in Chapter 4, a teaching company research project will be presented in next chapter to introduce the application of the proposed methodology and approach in a company.
6. TEACHING COMPANY RESEARCH PROJECT

6.1 Introduction

This research project was undertaken as a Teaching Company Scheme (TCS) supported by the Hong Kong Polytechnic University and Kingtronics Industrial Company Limited. The TCS aims to foster university-industry partnership. It supports local companies to take on graduate students pursuing a higher degree in local universities to assist in proprietary research and development work.

One of the major benefits of TCS is about applying academic knowledge for the direct, tangible benefit of business. It provides businesses with access to talented graduates, university expertise and government funding. Each partnership or TCS programme involves a graduate working on a project identified as central to the company's future commercial development. During the three years from 1996 to 1999, the author joined the TCS and was jointly supervised by the industrial and academic staff. The university provided teaching and research guidance to the author in this research project.

Through this arrangement, the company has benefited from the research results. On the other hand, the author has benefited from the practical research experience in a real industrial environment. The university has closer links with industry and a better understanding of its problems. In 1996, the author was appointed as the teaching company associate (TCA) to undertake the research project.
6.2 Overview

This project illustrates how the proposed methodology is applied in the web-based ERP system design and development. This project was implemented in the collaborated company by the author through the TCS as described in the previous paragraph, however, because of the confidentiality agreement, certain details of the system has been simplified.

6.2.1 Project timetable

The teaching company research project has taken three years to complete. The following timetable summarizes research work undertaken in these three years:

<table>
<thead>
<tr>
<th>Time</th>
<th>Tasks</th>
<th>DELIVERABLES</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>9/1996</td>
<td>LITERATURE REVIEW</td>
<td>List of previous publications</td>
<td>6 months</td>
</tr>
<tr>
<td>N/A</td>
<td>Industrial survey (Q2/97-Q4/98)</td>
<td>Ascertain the need of local enterprises in information technology for global manufacturing</td>
<td>3 months</td>
</tr>
<tr>
<td>6/1997</td>
<td>Evaluation of existing design methodologies and feasibility study</td>
<td>List of market available solutions</td>
<td>3 months</td>
</tr>
<tr>
<td>1/1998</td>
<td>Construction of new methodology</td>
<td>Novel design methodology – Hierarchical Design Pyramid</td>
<td>3 months</td>
</tr>
<tr>
<td>4/1998</td>
<td>Detailed system design and analysis</td>
<td>Design report</td>
<td>3 months</td>
</tr>
<tr>
<td>7/1998</td>
<td>System implementation</td>
<td>Preliminary program coding</td>
<td>8 months</td>
</tr>
<tr>
<td>N/A</td>
<td>Publication of research outputs</td>
<td>Six (6) international journal papers, Two (2) conference articles</td>
<td>Continuous</td>
</tr>
<tr>
<td>1/1999</td>
<td>Prototyping</td>
<td>Software deliverable (WeRP)</td>
<td>6 months</td>
</tr>
<tr>
<td>1/2000</td>
<td>Testing and commissioning</td>
<td>N/A</td>
<td>3 months</td>
</tr>
<tr>
<td>3/2000</td>
<td>Writing up of thesis</td>
<td>Thesis</td>
<td>6 months</td>
</tr>
</tbody>
</table>

Figure 6.1: Teaching company research timetable.
A prototype of the system, implemented by the proposed design and development methodology outlined in the previous chapters, is one of the deliverables of the research project. In terms of academic contributions, several journal papers and conference articles (listed in Chapter 8) have been published to disseminate the state-of-the-art research outputs.

6.2.2 The collaborated company

The company described here was established in the year 1986. Over the past 15 years, it has become recognized as the most reputable Original Design Manufacturer (ODM) in the consumer electronic products business. The Head office of the company is located in Hong Kong and occupies 26,000 sq. ft. of office area housing over 100 employees where the crucial task of Marketing, R&D, Production Planning and Control, Financial Control and other functions are performed. The company maintains constant communication with its production plant and close relationships with its global customers. The company owns a professional R&D team, supported by advanced equipment for research, development and product design, to fulfill its customer needs and therefore launches numerous new products each year to satisfy market demand.

The plant of the company is located in the Special Economic Zone (SEZ) of Xiamen, which is one of several SEZs in China. The Science and Industrial Park of the company was built on 250,000 sq. ft. of land, which comprises three industrial buildings, one training building, one R&D building, and one administration building. Part of the buildings were completed during the period 1990 to 1995, with a total usage area of 500,000 sq. ft., being brought into use since 1990. The company has
over 3,000 workers and staff in Xiamen. Several million telecommunication and audio products are produced and sold every year. The company is one of the biggest telephone answering machine manufacturers in the world, as well as one of the hundred top export manufacturers in China.

The company is a very typical Hong Kong based manufacturing company, which started its business in ODM manufacturing at the very beginning, and shifted its focus to developing its own brand name worldwide.

6.3 Statement Of The Problem

As stated in the previous paragraph, the company started to develop its own brand name in the global market in 1997. The company identified the need that their internal information system must support the integrated electronic commerce application so as to provide the world-class service to their global customers.

In fact, the electronic consumer product industry is demanding and dynamic in nature. Customers expect better product and services, ever-increasing product variety, and all in a shorter space of time. In addition, there is a profound shift in the way product is sold, not least because of the impact of the Internet. Against this background, the company must ensure efficiency, flexibility and transparency, even beyond the borders of the organization, throughout the extended enterprise – from suppliers, to distributors and retailers, right the way through to the consumer. This initiative strengthened the determination of the top management of the company to revolutionize the existing global information infrastructure.
The company was using several different applications to provide information internally and to its customers. Financial and manufacturing packages from diverse vendors were linked to a custom-built order management system. The batch interfaces and replication of data within the different applications were creating problems in terms of reconciliation and timeliness. In addition, the company found that there was a limit to how quickly and accurately they could respond to the customer’s order, because all the systems were so different and it was not practical to give the service people access to the entire database.

All in all, the company needed to take an information system environment beyond planning to execution. Moreover, it required leading edge data warehousing, available-to-promise and electronic commerce tools. Only then would the company achieve competitiveness and profitability. The following sections describe how the company applied the proposed HDP methodology (Figure 4.1) to develop a new system which enables the company to create a highly integrated solution geared precisely to their processes. The methodology applied is ideal basis for synchronizing the supply chain – because it is open, flexible, and designed to support the latest Internet technology.

6.4 Design And Analysis – HDP

Applying the HDP methodology, three critical steps must be undergone, namely (a) Business Process Reengineering, (b) Structural Analysis, and (c) Object Model Design. The following sections illustrate the application of the HDP through this teaching company research project to develop an e-commerce enabled ERP system.
6.4.1 Business process reengineering

In the previous section, an implementation paradigm for ERP and BPR integration with four different dimensions is proposed. The author identified that embedded BPR should allow the setup of workflow according to the organization processes and user requirements. In order to conduct business process reengineering (BPR) it is necessary to have a planning process. The process should describe the steps to be undertaken, who should be involved, what information is needed, and what are the outputs. This process should be simple enough so that it can be easily followed. From 1996 onwards, the company has identified the process for developing business strategy. Essentially it follows four major phases, they are (a) Establish the present position (b) Analyze strategic requirements (c) Develop strategic improvements (d) Formulate implementation strategy. This process is conducted by a team consisting of the general manager, manufacturing manager, plant managers, e-business manager and personnel from engineering, quality, marketing, human resources and information system divisions. In addition, a consultant is employed from outside to act as the facilitator who provide the experience, the methodology and to ensure that direction is correct and consistent. Much of the planning is done in small group meetings which allow time for interchange of ideas and discussion. Frequently, the managing director is involved to establish his views concerning the implications of the business strategy for manufacturing. The business process become apparent when going through the steps, they are addressed as follows:

(i) Global manufacturing strategies are needed to respond to business strategies
(ii) Comparison of present levels of manufacturing performance with future requirements

(iii) Strategies that remedy present weakness and exploit strengths

(iv) Strategies to cope with the political, economical and social environment

(v) Competitive advantages to sustain business growth.

Figure 6.2: Global strategic/manufacturing operation.
Figure 6.3: Global manufacturing strategy development and implementation.
Figure 6.2 illustrates the strategic manufacturing operations in Hong Kong and China that links the business objectives to manufacturing. Figure 6.3 illustrates the four steps in the manufacturing strategy development process and implementation.

(A) Establish present business and operations perspectives

The first step in the process is to identify the present business and operation position as illustrated in Figure 6.2. The following summarizes the major findings in step one.

(i) Research and development

The R&D department consists of 20 project engineers in Hong Kong and 50 in China. The industrial design and specifications are in general provided by the customers and the main task of R&D is to conduct research of new technology, develop new product according to customer needs and support manufacturing. Their major functions are (a) collect and conduct research of new technology, try to commercialize products (b) products design including the electronic and mechanical drawings, specifications and build engineering samples as well as the BOM (c) issue the engineering changes (d) authorize part approval (e) release engineering work instruction (f) design of new production method.

(ii) Information system

Current information system supports around 30 terminals in Hong Kong. However, the system was designed ten years ago and is considered insufficient to meet the current business requirements. In particular, the users in China office cannot access the real-time information of the company. The batch interfaces and replication of data within the different applications were creating problems in terms of reconciliation and
timeliness. Regarding the order management system, important features such as “Available to promise”, multiple part number and vendors, alternate parts, lot traceability and stock history are not available. Therefore, the salesman does not have current and future inventory information or credit information immediately available when the order management system was taking an order. The level of customer service has been affected substantially.

(iii) Competitors

Price, Quality and Delivery are the main competitive criteria of the company to win orders. The company is able to maintain good quality, well-established relationship with customers, on-time delivery and reasonable price to be competitive. However, it is under increasing pressure for shorter lead-time to delivery, wider product range and smaller batch size. Although sales turnover has increased, profit margin has suffered which indicates that cost reduction is necessary to be price competitive.

(iv) Quality and vendors

The ISO9001 provides a framework for the company to maintain a quality standard that is acceptable to the customer. A system of quality manual, documentation and procedures are available to ensure work instructions, inspections and test plans are followed. The system is implemented in various departments to ensure product quality. Production scrap rate is set at 1% and current scrap rate is estimated to be 0.5%. Overall quality is considered acceptable and meets customer specifications. Vendor purchasing is one of the major functions of the material department. Quality of vendors is evaluated through Incoming Inspections, documentation, procedural
control, and on the daily Material Review Board meeting. Current vendor delivery performance is acceptable.

(v) Production capability and capacity

The production control department is responsible for production planning and control. A "Production Plan" is prepared according to income orders and issued for production. Production Plans are scheduled for production using standard time estimated from the manufacturing engineering and the material status from material control department.

(vi) Costs and delivery

Product cost can be apportioned as around 60%-85% of material and 15%-40% of labor, overhead and profit which depends on difference products. The net profit margin is decreasing year by year because of the increase of operation cost. The current delivery performance is considered to be around 95% on time for mature products. New product delivery performance is however less satisfactory during the learning period in beginning of production. Lead-time from order confirmation to production required 10-12 weeks which normally consist of 4 weeks of production plus material acquisition. Customer in general would ask for shorter lead-time for order processing and inquiries. Customer order changes are frequent and the company normally requires four weeks advance notice.

(B) Strategic requirements

The second stage of the strategic planning process is illustrated in Figure 6.2. The company is now able to identify its competitive edge and remedy its weakness. The vision of the company is to produce high quality branded products with competitive
price. Hence, it plans to increase productivity through better use of information system, logistics, management and technological skills of the company. The main sustainable competitive advantage of the company is its strong customer base and good quality product. Understanding of the engineering and manufacturing processes to support mass production also contributes to the success of the company. However, the competitive advantages can be eroded by the considerable downward pressure on prices, increasing difficulties in planning and control of logistics, inadequate information system. The strong customer base and good image of the company have provided ample opportunity for the company in extending its product range and market. To realize the opportunity, the response to change in product design, engineering, manufacturing processes, logistics, workforce flexibility, and hence management style need to be strengthen. The main strategic issues to support its competitive advantage, hence increase in productivity is identified as logistic system to coordinate, plan and control material and manufacturing in Hong Kong and China. This system should encompass web-based ERP system with real time supervisory monitoring, control and information management applications.

(C) Develop strategic improvements

In step three of the strategic development processes, the company examined the major strategic improvement areas. These are summarized as follows:

(i) Production/material control

In overcoming the major problems mentioned earlier, real-time monitoring and ERP systems play a vital role in the strategic improvement. It will effectively integrate the logistics of material procurement, inventory, issue and tracking of production orders.
(ii) Shop floor

Quality has been monitored through in-process and final inspections. Re-work areas, WIP and raw material areas have been reserved. However, the use of shop floor data acquisition system using bar-code or industrial sensors to monitor shop floor status such as raw material, flow situation, quality inspections, etc. could effectively speed up the management of shop floor and use of equipment and manpower, and increase in productivity. The development of a real time monitoring system is important to management to identify the up-to-date machinery health status and trending/prediction reports to determine machinery maintenance.

(iii) Warehouse

It is important that material stores and areas are clearly identified and access restricted. Location and balance of the stock items should be accurate, transaction type and procedures clearly defined. Again the real time monitoring system and ERP system with stock control function will help trace all in and out transactions on material and hence the reasons of material shortage can be identified and corrective action taken. Bar-code-based information system is useful which records stock transactions on receiving and issuing and provide planning and control information to warehouse management.

(iv) Information system

Current information system is inadequate to support the business, particular with the expansion of manufacturing facilities in China and overseas sales office in the UK. The strategic issue is to coordinate the China and Hong Kong plants and thus eliminate or reduce the problems of fragmented supply-chain network and the
dissipation of resources on areas which may be marginal or even damaging when viewed from an overall business perspective. An effective global information system will reinforce the communication and interdependencies between the manufacturing, marketing and logistics.

(v) Logistics function

It is apparent that the logistics functions will become increasingly complicated in the future. With the global competition, manufacturing plants and distribution networks in Hong Kong and China, the company needs to develop a planning and control system that accounts for different customers, vendors and infrastructure of the manufacturing environment. The physical separation of the Hong Kong headquarters and its production factories in China bound to create many logistic problems. An effective real time monitoring and ERP system will eliminate some of the problems, however, the company needs to ensure that the plants are made use of their corresponding competitive advantages and that they are coherent to the company's business objective.

(D) Formulate implementation strategy

Of the strategic improvements discussed, the immediate actions are prioritized as follows:

(i) Implement of web-based ERP system

(ii) Integration of e-commerce application to the ERP system
Detail programs on the selection of hardware and software, forming of management committees are established to implement these strategies. Programs are formulated for the next three years. Specific actions are defined, responsibilities and authority are given to individuals for accomplishment and resources are identified for programs along with a time schedule. To implement the manufacturing strategy regular evaluation and review is required. A quarterly review session is scheduled. Action programs are evaluated for progress and possible corrective action. An annual review is also conducted to review the entire manufacturing strategy. They form the regular part of management of the company. The following sections detailed the system design, analysis and implementation based upon the proposed model HDP (see Figure 4.1).

6.4.2 Structural analysis - IDEF model

Taking the benefits described in the previous chapters, the implementation of the web-based ERP using HDP approach has been carried out in the collaborated company after the business strategy had been formulated as described in Section 6.4.1. Due to confidentiality agreement, this section only gives an extended summary of the case study to illustrate the concepts and functions of the proposed methodology in developing a global manufacturing information system.

In the context of this case study, web-based ERP is a corporate-wide initiative that includes teams from Corporate, Business Units, and Marketing Regions. It is the implementation of a common set of processes, applications, and performance measurements for the corporation. Therefore, it is the management of all interactions
involved in the initialization of an order to the delivery of that order to the customer.

The main objectives of the system are:

(i) Reduce Costs;

(ii) Speed up product delivery;

(iii) Guarantee common performance measurements across the Corporation;

(iv) Shorten development time;

(v) Create a collaborative culture to fully leverage Corporate’s global resources;

(vi) Enable the Corporate organizations to quickly adapt to global market dynamics.

In the HDP model, the first step to initialize the development is to construct the IDEF0 functional diagram. Figure 6.4 shows the context of the development of the web-based ERP system for the collaborated company. The descriptions of the activities involved are listed below.
Figure 6.4: Development of the web-based ERP system.

(A) Analyze and identify requirements (A31)

This is a very important area of HDP since all parties concerned must be certain what they wish the ERP system to achieve. This process of specifying the requirements can be quite exhaustive, but its importance should not be underestimated, considering the work and expense that will be incurred following the requirement definition.

(B) Analyze and review the current system (A32)

The system designer studied as much as possible into already existing systems, in order to avoid spending time developing something that is already commercially available at a reasonable price and to learn about design ideas he has not previously thought of.
(C) Initiate ERP design (A33)

When the preliminary work is completed and the requirement model generated, we cannot begin the system design phase at this stage. Similar to the project model required during the initial organization, the design phase must first be properly initiated. The decomposed page of Node A33 is shown in Figure 6.5. The activities involved in the node A33 are design generic ERP functions, design system database structure and logical process design and the details as listed below.

![Diagram of Initiate ERP design (A33)](image)

Figure 6.5: Initiated ERP design.

(1) Design generic ERP functions (A331)

Some of the functions, which are critical to an effective ERP system, and must be addressed, are shown in Figure 6.6. These system functions are listed in the node tree. Two functional designs (Design planning system and design production system) are illustrated in this section.
Figure 6.6: Design ERP functions.

Design planning system (A3311)

This includes three planning systems which are the Design Master Scheduling Planning System, the Design Capacity Requirements Planning System, and the Design Material Requirements Planning System. The design criteria and the interrelationship between these systems are described as follows:

(i) Design Master Scheduling Planning System

The function of this system/module is to help the master scheduler develop a realistic production schedule that provides a level plant load. This helps to develop a plan of resources and compares it with the resources available. As a result, inventory investment can be minimized by matching the master production schedule against actual and forecast demand. Customer service can be facilitated by providing
accurate, time-phased production in accordance with the guaranteed schedule. For the sake of a more realistic production plan, the examination of critical resource availability against requirements is also made possible.

(ii) Design Capacity Requirements Planning System

In the capacity requirements planning (CRP) system, some interactions can be identified. To cite an instance, work order control and production activity control provide the time phased work order input needed to project the work centre load from work-in-process. In addition, the planned order input for manufactured MPS items in CRP is supplied by the master scheduling. On the other hand, the material requirements planning module provides the planned order input for manufactured items from regenerative or net change MRP schedules.

(iii) Design Material Requirements Planning System

The purpose of this system is to provide the information and control for the company to effectively plan and manage the production and execution of the sales forecast and master schedule. The information derived from this system not only indicates the material requirements and order processing necessary to meet a company’s plan, but also assists management in planning and monitoring cash flow. Specific management objectives are to (a) minimize the amount of inventory on-hand and on-order to support the master production schedule, (b) plan purchases and production with the correct timing and priority, and (c) determine the cost impact of executing the master production schedule.
Design production system (A3313)

The production system is one of the essential ingredients of ERP system. The generic design and their functions are described as follows:

(i) Design Job Order Control System

The aim of this module is to allow the user to tie production activity and costs to a job. Production budgets and job estimates can be built from the cost data defined for the components and assemblies to be used in the job. Detailed cost reporting can be provided to monitor all costs charged to the job. Specific management objectives are to (a) provide realistic job cost budgets and estimates, (b) control the costs of jobs, (c) monitor the progress of jobs through the production processes, and (d) provide actual cost variance reporting.

(ii) Design Work Order Control System

The work order control system is used to provide a complete and comprehensive method for production costing analysis. It can easily be structured for multiple levels of manufacturing which involve tracking from the component part level to the finished product. In addition, it tracks the various operations, procedures, and labour.

(iii) Design Production Activity Control System

Work Order Control schedules open operations from work orders while Capacity Requirements Planning uses shop scheduling and the planned orders from MPS and MRP systems to calculate the capacity plan and capacity load. Therefore, the production activity control system enables the user to monitor the progress of
operations through the production facility. Dispatch reports can be produced using several priority methods.

(iv) Design Purchasing/Receiving System

The purchasing/receiving system maintains information about approved vendors and approved manufacturers, purchase orders for multiple parts and multiple scheduled deliveries per part, receiving, receiving inspection, vendor shipments, and buyer performance. This system accumulates information for both purchasing and vendor performance evaluation. Vendor information is placed into the purchase order. The purchase order data are placed into the receipts and, finally, the receipt information is transferred into accounts payable receipts and material inventory movements.

(v) Design Total Quality Control System

Total quality control is particularly important to the design of the ERP system which also forms a part of the TQM process. Specific management objectives are to (a) control incoming and in-process material inspection, ensure that material is inspected to proper inspection criteria and the correct sampling plan for the part or vendor, (b) maintain current inspection specifications and sample plans for use in inspecting material, (c) manage vendors through approval for supplying parts and certification allowing the inspection to be by-passed, and (d) allow vendor selection through vendor analysis based on quality, price, and delivery performance. All the management objectives are well documented in the quality system manual of the company.
(vi) Design Inventory Management System

This system is designed to handle three types of items namely (a) manufacturing items, (b) trading goods and (c) fixed assets. The design of this system seeks to maintain accurate information on the quantity, location and value of all inventory items. The aims are to reduce inventory investment and handling costs and to improve customer service through better delivery schedules. By improving the integrity of inventory data, the basis for better material planning and control is established. The system specifically aims to (a) increase inventory turnover and reduce inventory investment with a corresponding reduction in handling costs, (b) improve customer service through shorter lead times and reduce late customer deliveries, thereby improving a company’s competitive position, and (c) improve the integrity of inventory on-hand data and provide a basis for better material planning and control.

(2) Design database structure and system evaluation (A332)

Referring back to Figure 6.5, the database structure is the vital part of the ERP system in which systems must interface with each other, in order to provide critical information. For example, the inventory database must be updated as soon as components are withdrawn from or returned to stock. The work in process (WIP) system must interface with the inventory database to track the location and the status of assemblies in storage. Process planners should access the inventory database to ensure that only materials available are assigned for production. Dynamic object diagrams were introduced to capture the flow of information, according to the HDP model, because IDEF0 cannot capture the information flows between several functions simultaneously. An object diagram is stored in the data store which shows the data movement within the system and yields a set of interested data items. These
items are logically related to one another. The relationship between these data items is modeled by an object model diagram, which will be introduced in the next section. After the consolidation of the object model, mapping of the IDEF0 model to the relational model can be carried out.

(3) Design logical process (A333)

Referring back to Figure 6.5, this process outlines in detail the operations that the system performs either in response to an enquiry or retrieval. The software programme is implemented based on the process outlines structured by adopting the OOM as stated in the HDP model.

(D) Test and analyze new system (A34)

Referring back to Figure 6.4, different proposals of ERP are evaluated during this stage. To accelerate such a time consuming and costly process and promote accurate performance data, simulation is recommended. In the evaluation process, the system design should not overlook the compatibility of the ERP system with other information systems currently running in the company. A final decision can then be made as to which alternatives are to be chosen and implemented.

The structural analysis of the new ERP has been completed in this phase. The forthcoming sections are based on the outputs of the structural analysis to outline the object design of the system. Through this HDP methodology, time and human resources can be minimized to achieve a novel architecture of the web-based ERP system.
6.4.3 Object model design

The information captured in the IDEF0 function model in the previous section can contribute to the object model analysis. IDEF0 provides valuable insights into initial classes and routines that the system requires. Thereafter, two objects, namely customer object and order object, are introduced in this report so as to elaborate the application of the proposed HDP methodology. Referring to Section 4.4.1, the object model encompasses two parts including hierarchical static model and ladder dynamic model. The hierarchical static model specifies the hierarchical structure of the static design objects. The ladder dynamic model specifies communication between objects and transitions between states in a ladder-liked format.

(A) Customer object

(1) Formulation of hierarchical static model

![Diagram of object model]

Figure 6.7: Object model of “People” and “Customer”.

Figure 6.7 shows the object relationship of the objects in Booch notation. The following tables illustrate the design schema of superclass “People” and subclass “Customer”. The two objects demonstrate an “is_a” relationship. The attributes of
“People” are also the attributes of “Customer”. To identify all objects in a system from scratch is not an easy task. Through the HDP methodology, objects and their attributes can be generated from the IDEF0 conceptual diagram.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email</td>
<td>Text</td>
</tr>
<tr>
<td>Firstname</td>
<td>Text</td>
</tr>
<tr>
<td>Lastname</td>
<td>Text</td>
</tr>
<tr>
<td>Address</td>
<td>Text</td>
</tr>
<tr>
<td>City</td>
<td>Text</td>
</tr>
<tr>
<td>State</td>
<td>Text</td>
</tr>
<tr>
<td>Country</td>
<td>Text</td>
</tr>
<tr>
<td>Phone</td>
<td>Text</td>
</tr>
<tr>
<td>Passcode</td>
<td>Text</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company</td>
<td>Text</td>
</tr>
<tr>
<td>Credit</td>
<td>Text</td>
</tr>
<tr>
<td>Bill Address</td>
<td>Text</td>
</tr>
<tr>
<td>Bill City</td>
<td>Text</td>
</tr>
<tr>
<td>Interests</td>
<td>Text</td>
</tr>
</tbody>
</table>

Figure 6.8: Database schema of “People” and “Customer”.

The super-class People inherits a subclass – Customer. They follow the “IS_A” relationship of object-oriented design principle. The attributes of the Customer object are inherited from the super-class People. The Customer object contains three major operations, including authenticate, register and update a user, which will be further discussed in the following section.

(2) **Formulation of ladder dynamic model**

The ladder dynamic model specifies communication between objects and transitions between states in a ladder-like format. In this case study, the authentication process of a customer will be demonstrated. The Customer object mainly performs three functions:
(i) Authenticating existing customers. In doing this, it will query the customer database to see if a customer matches the supplied customer ID and password.

(ii) Registering new customers. The object will take the supplied customer information and insert it into the database.

(iii) Updating of an existing customer’s information.

The business logic behind authentication of existing customers and registering new customers will be encapsulated in this object. The graphic presentation of ladder diagram is depicted in Figure 6.9.

![Ladder diagram of user authentication](image)

Figure 6.9: Ladder diagram of user authentication.

(B) Order object

(1) Formulation of hierarchical static model

Figure 6.10 illustrates the formulation of the static model of the “Order” object. The attributes and behaviors of the Sales Order and Purchase are inherited from the superclass Order. The sales order is further elaborated to demonstrate the concept.
Figure 6.10: Object model of “Order”.

The following table illustrates the design schema of superclass “Order”.

<table>
<thead>
<tr>
<th>Order Processing</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td></td>
</tr>
<tr>
<td>OrderID</td>
<td>AutoNumber</td>
</tr>
<tr>
<td>Email</td>
<td>Text</td>
</tr>
<tr>
<td>PaymentMethod</td>
<td>Text</td>
</tr>
<tr>
<td>OrderAmount</td>
<td>Currency</td>
</tr>
<tr>
<td>ShipAddress</td>
<td>Text</td>
</tr>
<tr>
<td>ShipCity</td>
<td>Text</td>
</tr>
<tr>
<td>ShipCountry</td>
<td>Text</td>
</tr>
<tr>
<td>PromisedShipDate</td>
<td>Date/Time</td>
</tr>
<tr>
<td>ActualShipDate</td>
<td>Date/Time</td>
</tr>
<tr>
<td>Fulfilled</td>
<td>Number</td>
</tr>
<tr>
<td>BasketCreation</td>
<td>Date/Time</td>
</tr>
</tbody>
</table>

Figure 6.11: Database schema of “Order Processing”.

The sales order object is adopted in the electronic commerce application in sales and marketing sub-system discussed in Figure 6.6. As the user proceeds through the system and adds selections to their electronic “shopping cart” – the physical representation of the sales order object, that information is tracked by the system through the object. The data consists of the quantity of each item that the user has
identified. Once the user has selected all of the items that they are interested in, the content of the shopping cart then becomes the order information.

(2) Formulation of ladder dynamic model

The Order objects are collectively responsible for the important phase of the application. Figure 6.12 illustrates the high level system flow of the order processing procedure. The objects are used to take the contents of the basket and turn them into a sale. These objects have four major pieces of functionality to support the order processing function.

(i) The Order object is used to compute the pricing information for the order. Its responsibilities include determining the selling price of each item in the basket. The selling price is determined by modifying the list price by any promotions that may be in effect at the time of purchase. It then calculates the total price of all items in the order. Next, it computes the shipping costs and the sales tax according to the global tax reference engine. Finally, all of these figures are added together by the object to arrive at the total cost for the order.

(ii) The Payment object is responsible for validating the payment method that the users selected. This object can use multiple methods for validation, based on the method of payment being used.

(iii) The Inventory object is responsible for maintaining the proper inventory information for all of the items in the database. The object will deduct from the database the quantity of items ordered in each order. The object could also be
responsible for sending notification if the inventory of an item drops below a certain threshold.

(iv) The Shipping object is responsible for the processing of all of the shipping information for the order. This object is used to transmit the shipment information to the shipping agent. It is responsible for determining the date that the order will ship and estimated arrival date for the shipment.

![Ladder diagram of order processing over the website.](image)

Figure 6.12: Ladder diagram of order processing over the website.

### 6.5 Implementation - WOOM

#### 6.5.1 Physical architecture

In the teaching company research project, the physical architecture implemented by the author consists of two web servers and two database servers connected using
Microsoft Cluster Server (MSCS) that share replicated information hosted at the corporate office in Hong Kong. The physical architecture is shown in Figure 6.13.

Figure 6.13: Physical architecture of the prototype system.

Clustered SQL servers are centralized database of the entire organization. The clustering service provides the highest availability of the system to support the global service. In order to support the original information system and manual procedure, an electronic data interface (EDI) engine has been implemented to cater the hybrid automation environment.

6.5.2 Logical architecture

The whole web-based ERP system is built on a foundation of Microsoft Active Server Pages (ASP), Microsoft Internet Information Server 4.0, Microsoft Windows NT® Enterprise Edition and Microsoft SQL Server 6.5. Its functionality was fashioned
before Site Server Commerce Edition became available. Figure 6.14 shows the logical architecture for the application.

![Diagram of logical architecture](image)

Figure 6.14: Logical architecture of the prototype system.

### 6.5.3 WOOM architecture

WOOM is an important component of our proposed methodology. In this stage of implementation, the application architecture and the data architecture have been developed based on the WOOM architecture. There will be four domains in the system: Logic Domain (LD), Presentation Domain (PD), Database Domain (DD) and Services Domain (SD). Figure 6.15 illustrates the overall picture that the author has developed in this teaching company research project.
Figure 6.15: Overall WOOM architecture.

A lock symbol can be found in the diagram. All data communications in this architecture are supported by full encryption in order to protect the privacy and data integrity of the whole system. As there are limited resources in developing the prototype system, some state-of-the-art security measurements have not been incorporated. Therefore, suggestions in security can be found in the Future Work (Section 8.3).

(A) Logic Domain

Logic Domain is used to encapsulate the business logic of the system and is called directly by the Services Domain code. It is responsible for manipulating the data in the system to provide the user with the proper information. The parameter that is passed to the method is a collection, as is the value returned by the method. In the following example, the InsertCustomer method is introduced in Figure 6.16.
Eventually, the information for the new customer is passed in as a parameter. This parameter is a collection of information that the system has gathered from the user. At this point in the application, the user has entered their customerID and password. This information has been passed to the SelectCustomer method, which was unable to locate the matching customer record and so returned False to the application. The application can then call the InsertCustomer method to create a new customer record in the database.

```
Public Function InsertCustomer(ByVal vCustData As Variant) As Variant

  oRs.AddNew
  oRs.Fields("Email") = LCase(vCustData("email"))
  oRs.Fields("Password") = LTrim(vCustData("password"))
  oRs.Fields("Firstname") = ""
  oRs.Fields("Surname") = ""
  oRs.Fields("Address") = ""
  oRs.Fields("City") = ""
  oRs.Fields("Country") = ""
  oRs.Fields("Phone") = ""
  oRs.Fields("UseCookie") = "N"
  oRs.Fields("Interests") = ""
  oRs.Update

Exit Function

InsertCustomerErr;
ObjectContext.SetAbort
Set InsertCustomer = RaiseError ("InsertCustomer")

End Function
```

Figure 6.16: Sample code in Logic Domain (LD).

The logic domain is the major building block of the implementation. In this example, an AddNew method is illustrated to create a new record. Since the only fields that have data in them are the user’s email address and password, all of the other fields
would be initialized to their default values. The Presentation Domain would use another method to set the presentation of remaining fields in the record. The following section is going to demonstrate the actual implementation.

(B) Presentation Domain

Presentation Domain is mainly concentrated on the manipulation of Web-based user's interface and interaction with the user, i.e. the Web browser. In order to initialize the Presentation Domain, three important parameters have been passed to the PD object. Those parameters are the user interface UIOption, appearance of the control Appearance, and the initial background color BackColor of the control. Figure 6.17 illustrates some of the implementation.

```
<table border="1" cellpadding="1" cellspacing="1" width="100%">
  <tr>
    <td><font color="purple" face="Arial" size="1">Description:</font></td>
    <td><input type="text" class="Readonly" readonly id="item_desc" maxLength="80" name="item_desc" size="50" style="HEIGHT: 20px; value="" height="20"></strong></td>
  </tr>
  <tr>
    <td><font color="purple" face="Arial" size="1">UIOption</font></td>
    <td>Value="2047"</td>
  </tr>
  <tr>
    <td><font color="purple" face="Arial" size="1">Appearance</font></td>
    <td>Value="3"</td>
  </tr>
  <tr>
    <td><font color="purple" face="Arial" size="1">BackColor</font></td>
    <td>Value="255"</td>
  </tr>
</table>

Figure 6.17: Sample code in Presentation Domain (PD).
```
In this research project, the author employed the standard 'default' values. A UIOption of 2,047 displays the control in its default configuration with all of the features enabled. An Appearance setting of 3 displays the control with a three dimensional appearance, and the value of 255 for the BackColor sets the color to red.

(C) Database Domain

Database Domain (DD) is a standard interface for accessing the physical databases. It is the major interface in the system that is responsible to retrieve, manipulate and update data directly to/from the database system. It is used to encapsulate the database access of the system. The Logic Domain can only interact with Database Domain. The instances in Database Domain cannot be directly accessed by other domains. These instances would directly manipulate that information in the database as well as managing the connections to the database. There are three instances in this system.

In order to illustrate the implementation, a DDCustomers Object is introduced, which is responsible for managing the data connection between the LDCustomers Object and the database. This object supports the retrieval of customer information, the addition of new customers, and the modification of existing customer information. There are three methods that the DDCustomers Object supports for manipulating the data in the Database Domain. These methods allow users to insert a new customer, select an existing customer, and update the information of an existing customer. Figure 6.18 is the illustration of the implementation.
Set cnn = Server.CreateObject("ADODB.Connection")
cnn.Open Application("Connection")

On Error GoTo InsertCustomerErr
Dim vResponse As New Collection
Dim vCustData2 As New Collection
Set ors = oObjectContext.CreateInstance("ADODB.Recordset")

ors.open "Customers", vDbConn, adOpenKeyset, adLockPessimistic, adCmdTable

Figure 6.18: Sample code in Database Domain (DD).

The Recordset would be initialized by using its open method. Since the database connection has not been defined, we pass the vDbConn variable which is a global variable that holds the database connection string. Since it adds information to the database, and do not need to access new information from other users, the adOpenKeyset flag would update the recordset only with new records that users add. The adLockPessimistic flag would cause the record that the recordset is currently working to be locked at all times.

vCustData2.Add Trim (ors.Fields("Email")), "email"
vCustData2.Add Trim (ors.Fields("Password")), "password"
vCustData2.Add CStr (ors.Fields("CustomerID")), "customerid"
vCustData2.Add "", "firstname"
vCustData2.Add "", "surename"
vCustData2.Add "", "address"
vCustData2.Add "", "city"
vCustData2.Add "", "state"
vCustData2.Add "", "zipcode"
vCustData2.Add "", "country"
vCustData2.Add "", "phone"
vCustData2.Add "N", "usecookie"
vCustData2.Add BuildInterests(""), "Interests"

Figure 6.19: Sample code in Database Domain (DD) for adding a new customer.
Once the Update method of the recordset is called, the new record is added to the database. Since the CustomerID field is automatically generated after a new record is added, we can get its value from the recordset and pass it back to the application in the response collection. Figure 6.19 demonstrates how to search for a customer in the database and add a new customer to the database.

(D) Services Domain

Service Domain (SD) is consisted of two agents; they are the Supply Agent and the Demand Agent. The Supply Agent is an external communication interface that enables the Logic domain to release its public services for another WOOM object. Another object can request service from the Supply Agent of any object through its own Demand Agent.

In the following example, we create the response collection by a pair of Supply Agent and Demand Agent. Since the majority of the fields have no data, we just add blank strings from those entries in the collection.

```plaintext
.. supply.vResponse.Add False, Key:="error"
supply.vResponse.Add True, Key:="exists"
supply.vResponsomes.Add vCustData2, "custdata"

oRs.Close
oObjectContext.SetComplete
Set InsertCustomer = vResponse
```

Figure 6.20: Sample code in Service Domain (SD).
Figure 6.20 illustrates the communication between two WOOM objects by a pair of Supply Agent and Demand Agent.

### 6.5.4 Graphical user interface

Figure 6.21 shows the logon screen of the prototyped web-based ERP system implemented using the approach described earlier. In this thesis, inventory management system is used to illustrate the prototype. The prototype was tested and implemented in the collaborated company. This screen serves as a user verification gateway for the whole system. In this prototype, it only supports the English language.

![Logon screen of web-based ERP system](image)

**Figure 6.21:** Logon screen of web-based ERP system.
Figure 6.22: Major functions of web-based ERP system.

After logging on the system, seven sub-systems can be seen (Figure 6.22). The system can be divided into five functional areas, namely, Planning, Costing, Shipping, Engineering, and Tooling. A report generator is a centralized workspace for managers to get the inventory positioning of the company. User manager is a tool for the system administrator to set up the user security access right of the system.

Figure 6.23 demonstrates all the reports found in the system. This system can be accessed by three different parties including UK office, Hong Kong headquarter and China Vendors.
Figure 6.23: Report generator of web-based ERP system

A selection box can be found in the most upper right-hand corner. This box serves as a shortcut to different functional areas. This function is supported by the WOOM architecture.

Figure 6.24 illustrates the ad-hoc results in one of the functional areas – engineering department. The information can be searched by item, description, or purchase order number. For the engineering department, only the purchase order number can be manipulated, which is the business logic defined in the BPR stage of HDP methodology (Figure 4.1).
### 6.6 Implementation Considerations

This research has significant implications for companies that have implemented or are about to implement ERPs. ERP projects are often associated with fundamental organizational improvement efforts, such as business process reengineering. Because ERP support business integration, they potentially represent more than a change in technical infrastructure. Indeed, the main benefits resulting from an ERP implementation may actually come from changes in the business, organizational structure, the roles and skills of organizational members, and knowledge member activities (Davenport, 1998; Martin, 1998).
A pilot run of the system was conducted for three months. The production line ‘XX’ was chosen for pilot run because this line was one of the typical production lines and only one product was involved: material flowed into the production line at low volumes from external suppliers and internal operation units. The purpose was to validate the design methodology as well as software deliverables. The following sections conclude the experience gained before and after implementation of our proposed HDP methodology.

6.6.1 The situation before and after ERP

Before the implementation of new global enterprise information system, the information system used by the company was a legacy system, i.e. a closed architecture with minimal flexibility. Section 6.4.1(A) has introduced the basic infrastructure of the information system in the company. Figure 6.25 summarizes the situations of before and after ERP implementation in terms of data information.

<table>
<thead>
<tr>
<th>TYPE OF DATA INFORMATION</th>
<th>BEFORE (TIME BETWEEN UPDATE RUNS)</th>
<th>AFTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic foundation data</td>
<td>Daily</td>
<td>On-line</td>
</tr>
<tr>
<td>- Inventory records</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Scheduled receipts</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Material list</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Item and work center</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Shop feedback</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operations plan</td>
<td>Weekly</td>
<td>Daily</td>
</tr>
<tr>
<td>Sales report</td>
<td>Weekly</td>
<td>On-line</td>
</tr>
<tr>
<td>MRP frequently</td>
<td>Monthly</td>
<td>Daily</td>
</tr>
<tr>
<td>Vendor schedule frequently</td>
<td>Weekly</td>
<td>On-line</td>
</tr>
</tbody>
</table>

Figure 6.25: Data information ‘before’ and ‘after’ of system implementation
It can be seen that the new system is now able to support the business practices both online and daily basis, while the old system was weekly and monthly. This change is a crucial factor in gaining competitiveness in a global manufacturing environment. It enables cost, delivery and quality to be attained with the highest customer satisfaction.

The previous legacy system was transformed into a scalable ERP solution that support with an open non-proprietary technology standards using Microsoft’s software solutions including SQL, ASP, Internet Information Server etc. The technology standards employed would provide a reasonable protection of investment and ensure minimal risk in the foreseeable future. The database now operates on most major client/server hardware platforms; Local Area Network and user interfaces; technology obsolescence and software upgrading are generally catered for. In fact, the detailed programming and technical knowledge skills are more widely available using the proposed design than before, it permits readily adaptation of the system in repose to ongoing changes in business and operation strategies and processes. Customization in terms of effort and time are much less than the legacy system before implementation.

6.6.2 Project team and training

In this research project, the prototype developed by the author was implemented in the collaborative company as an information infrastructure. In order to have a smooth implementation process, a steering committee has been formed which consists of Director, Project Manager, Executive Sponsor and Area Managers. In fact, much of the complexity during the implementation of our proposed HDP methodology has to do with setting of parameters such as accounting rules, operation and production workflow, designing menus and authorizations. It is therefore necessary to clearly
define the roles and responsibilities of different employees, understand and configured the system. The involvement and willingness of the employees to accept and use these new procedures as a result of our proposed global manufacturing information system would determine the success of the implementation. Eventually, the project team must not underestimate the difficulties of organization changes and implications of our new system which helps to support and manage dynamic business process in a global environment.

Specifically, the responsibilities of our steering committee include the allocation of resources (human and financial), conflict resolution, and formation of strategic decision relative to the project. The steering committee met at least once a week to review implementation status, performance, and budgets. The committee also resolves project team conflicts and establishes success criteria and performance measurement in project life cycle. Key project strategy questions for the Steering Committee are summarized as follows:

(i) which modules to implement first

(ii) which products lines to implement first

(iii) which divisions to implement first

(iv) timetables

(v) team and role assignment
(vi) coordination with other projects

During the implementation, the author was appointed as the project manager - the key individual in the entire project. The major responsibility of the project manager is to execute as well as to monitor the day-to-day project work. The managing director of the company has been acted as the executive sponsor who is responsible for initiating the implementation project, secures the commitment of human and financial resources, and recruits the essential project team members. In addition, project sponsor is the principal "change agent" behind the implementation.

Three area managers have been invited to participate in this project, including sales and marketing manager, factory manager, and finances manager. Their responsibilities are summarized as follows:

(i) Have functional area responsibility for implementation

(ii) Give input during prototype step

(iii) Must have sufficient authority in functional areas

(iv) Can be used to train others in their functional areas

(v) Provide input to implementation plan

(vi) Conduct training to staff of functional areas
6.6.3 Change management and leadership

Our proposed HDP methodology inevitably means new ways of adapting and operating. As discussed in the previous sections, the requirements of an information system require business re-engineering which is one of the core components of HDP model. The users including business executives and operation staff must be informed about the needs for such a change; from the company's perspective as well as from that of an individual. One of the tasks of the author as the project manager is therefore to train the team members and all other staff involved in the proposed design. In this respect, the author working with external consultants from the Hong Kong Polytechnic University and the Hong Kong Productivity Council have provided many inputs, high level as well as operational level trainings to support our implementation, and prevent any setbacks in our implementation. In particular, they have assisted to train the steering team of the process mapping and our proposed reengineering methodology - the concepts and tools. Moreover, they provide the training on detailed process analysis and redesign, project management techniques, leadership and decision making skills.

One of the most important successful elements gained from the implementation process is leadership skill. The author through the support of the managing director of the collaboration company as well as the consultants mentioned earlier, appreciated and applied this success factor. In fact, it is my experience that many executive of this company were supportive of our new design methodology but failed to provide constant real hand-on leadership and commitment because of the demand of time and effort they must put in. It is imperative for the author and managing director together
with the consultants to regularly stimulate the steering team to conduct process redesign, integration and business re-engineering.

All in all, change management and leadership is particularly important to the success of our project. During the implementation, the following strategies have been applied:

(i) A clear sense of mission or purpose is essential. The simpler the mission statement the better.

(ii) Build a team. Maintain a flat organizational team structure and rely on minimal and informal reporting requirements. In this project, a core project team has been built and described in Section 6.6.2.

(iii) Pick people with relevant skills and high energy levels. Functional managers in different departments were invited to take part in the project team.

(iv) Shift to an action-feedback model. Plan and act in short intervals. The total implementation cycle is not longer than three months.

(v) Keep the communications barriers low, widely spaced, and easily hurdled.

(vi) Deploy consultant to provide experience and leadership to support the project.
6.6.4 A good implementation methodology

In this research, the author proposed an innovative methodology to design and implement global manufacturing information system. To achieve the new business processes we have reengineered the existing processes in such a manner that many benefits have been gained. However, the experience gained from the implementation suggested that it is not essential to follow every step in the proposed methodology. The essential benefit of our methodology is to provide a road map to follow. The methodology enables the project leader to set out clear, measurable objectives at the very outset of the project, review the progress at intervals during the implementation processes. Our methodology is an integration approach; it is complicated and required attention on details. With many of the functionality and features of the new systems, a small change in the design can create a ripple effect across the entire system. The implementation methodology – WOOM – proposed by the author can provide the guidelines to determine the effort and time required for carrying out any of these detail activities.

Our ERP implementation has brought about many changes in the roles of different departments, requiring different skill-sets and changes in the authority and responsibility. In short it resulted in a change in the existing power structure, it is not only an IT implementation, it is an organization and business changes. Here, in this Teaching Company research project, the author and the managing director, together with the consultants and senior executives have been bold enough to face this challenge and commit to excellence. HDP methodology together with WOOM is the key success factor in ensuring successful implementation.
In this chapter, a case study in the form of teaching company research project was presented. Our proposed methodology was adopted and implemented in the collaborated company. A prototype and the implementation considerations are included in this chapter. The implementation results and discussion on the proposed methodologies will be presented in next chapter.
7. RESULTS AND DISCUSSION

7.1 Analysis Of Results

In order to evaluate the feasibility of the proposed HDP model in the global manufacturing environment, a prototype system was implemented in a manufacturing enterprise with head office in Hong Kong, production plants in China and sale channels in overseas. This enterprise information system represents a typical multi-mode manufacturing, multi-national as well as multi-plant production characteristic which is a typical global manufacturing operation. In this research, the HDP methodology has been adopted to develop such enterprise information system and the result is considered to be valuable. Two months after implementing the system, the company doubled the number of orders received over the Web compared with the original system. Currently, the Web is still the smallest percentage of their business (as compared to phones, faxes, and EDI), but as the system is further developed, the company anticipates it could become a larger portion of their business.

Besides, the benefits offered by the HDP methodology and the implementation process include management commitment, savings on overheads, an increase in productive hours and product quality, and enhanced participation and problem solving. Savings on overheads mainly include the elimination of an analysis report sheet, travel ticket, time sheet for payroll proposes, and the reduction of expenses on quality control, stock evaluation and payroll staff. The automation of stock evaluation now generates information on patterns of material consumption in a manner much more cost effective than it used to be. With real-time production information linked to
accounting and payroll sections, the cost incurred is now corrected accurately. Another remarkable time saving comes from error analysis which is now completely automated. These positive findings provide sufficient evidence to validate the proposed system.

![Customer Delivery Service Graph](image)

**Figure 7.1:** The customer delivery service performance for the year 2000.

![Quality Improvement Graph](image)

**Figure 7.2:** The quality improvement of product for the year 2000.
Figure 7.3: The production cost performance for the year 2000.

Figures 7.1 exhibits the improvement in customer delivery service. It illustrates the total number of promised shipments awaiting completion and the number of promised shipments completed during the month. After the enterprise information system is implemented, the number of promised shipments awaiting completion beyond promise due date, i.e. past dues, is reduced. Therefore, the customer satisfaction is improved. Moreover, Figures 7.2 and 7.3 describe the improvements on the product quality and production cost respectively.

7.2 Industrial Adaptability

A generic methodology to design and develop a global enterprise information system has been developed in this research project. The academic research result was validated in the industry through the Teaching Company Scheme. The scheme is co-operated by the Department of Manufacturing Engineering, Hong Kong Polytechnic
University and Kingtronics Industrial Co., Ltd. The collaborated company is a very
typical Hong Kong based manufacturing company, which started its business in ODM
manufacturing at the very beginning, and shifted its focus to developing its own brand
name worldwide. It is also amongst the 80% of companies, which have their product
plants offshore, and located in the most popular site in various part of China such as
Shanghai, GuangZhou, etc.

The methodology has been adopted by the collaborated company as the first step to
implement a global manufacturing model. Using the proposed methodology of the
author, the collaborated company has successfully implemented a simplified version
of a digital nervous system with users all round the world – Hong King, China and
England. This result is only possible with the joint research in the Teaching Company
Scheme. The cross-border barriers were solved by the proposed implementation
approach effectively. The collaborated company is committed to employ the
proposed methodology and further developed this project.

In addition, after three years of this research project at the collaborated company, in
1999, the author established a software company, focusing on web-based ERP system
development and implementation. The author found that the methodology proposed
in this research is valuable to the enterprise system development cycle. Several
industrial projects have been deployed to my clients using the HDP and WOOM
methodologies, and the success of my technology business has led to the award from
"Ten Outstanding Young Digi Persons" in 2000 organized by Hong Kong
Productivity Council, Hong Kong Junior Chamber, and Information Technology
Services Department of HKSAR Government.
7.3 Limitations Of The Methodology

The HDP methodology is a conceptual framework embedding BPR that helps manufacturing enterprises respond to dynamic changes taking place in a global business environment. WOOM is a model to provide a systematic framework for coding and programming in the actual implementation through Internet. In the information age, manufacturing companies are becoming increasingly customer focused. In fact, the migration towards customer-driven business models began before the advent of the Internet, but Internet technologies have dramatically accelerated the process. The following summarizes the major limitations of our investigation.

(i) Coding and enhancement

During the implementation phase, the proposed methodology might give rise to tight coupling, which is the major limitation. With strongly typed languages, such as C++, the compiler parses every class in the web to ensure type correctness. For the best scenario, this behavior may result in make-dependencies that cause a recompilation of the complete system after each minor bug fix. At the worst case, the compiler or linker runs against some internal limitation and refuses to build the program at all. Besides, it is difficult to isolate a group of classes for component tests. Tracking down errors will be a hard and time-consuming job, which is harder than any conventional implementation approach and the program may be hard to maintain. Changes propagate all through the system and it is difficult to determine the responsibilities of a class. Therefore classes blow up with dozens of attributes and methods.
(ii) Platform support

WOOM are best supported on Windows 2000 and XP platforms. However, WOOM over non-Windows platforms has few supporters. Until WOOM for alternate platforms has been fully tested, the technology is most appropriate in environments that are primarily Windows-based.

(iii) Platform specificity of WOOM components

Because WOOM is based on a native binary format, components written to these specifications are not platform independent. Thus, either they must be recompiled for a specific platform or an interpreter for the binary format must become available. Depending on user’s perspective, the use of a binary format may be either an advantage (faster execution, better use of native platform capabilities) or a limitation (ActiveX controls, unlike Java applets, are NOT machine independent).

(iv) Security

Because WOOM components have access to a version of the Microsoft Windows API, "bad actors" can potentially damage the user’s computing environment. In order to address this problem, Microsoft employs "Authenticode" which uses public key encryption to digitally sign components. Independent certification authorities such as VeriSign issue digital certificates to verify the identity of the source of the component. However, even certified code can contain instructions that accidentally, or even maliciously, compromise the user’s environment.

While I have described the technical limitations of the proposed methodology, there are management and organization constraints which affect the success of the
methodology; the existing information infrastructure, resistance to change, management commitment and leadership skill. These management considerations are fully described in the implementation section and therefore will not be elaborated here.

Moreover, our model has only been tested and proved to be successful through the teaching company partner. More effort and case studies are required to demonstrate that it can be applied to other industries. Further test of the applicability of this methodology in other similar industries and evaluation of the extent of customization are necessary to validate the robustness of our approach.

7.4 Benefits Of The Methodology

In the system design stage, the use of structural analysis methodology simplifies the stage of object identification. The use of logical model reduces the system complexity significantly. Moreover, applying inheritance and polymorphism theory in the logical model has enriched the inadequacy in the original object-oriented methodology. In short, this approach provides a consistent manner in all stages of development.

In general, the merits of the HDP methodology can be elaborated upon as follows:

(i) The approach can concentrate more on the whole picture, since the details of the problem are ignored by abstraction.

(ii) The HDP approach provides a mechanism to formalize the model of reality.
(iii) The design is more reusable and flexible, since the real world problem-domain is directly modelled out and the physical changes in the system do not make the design change.

(iv) The development is done recursively from a small part to a whole system, so large-scale integration is seldom needed during the development cycle.

(v) The HDP approach is more suitable for handling a large group of similar problems than the object-oriented approach solely, since the IDEF development can effectively refine the essential entities in the problem-domain.

(vi) The implementation time is the shortest in the HDP approach.

A table of comparison is shown in Figure 7.4 in order to justify the methodologies discussed.

<table>
<thead>
<tr>
<th>Considerations</th>
<th>Abstraction/Information hiding</th>
<th>Natural way of thinking</th>
<th>Reusable/Flexible</th>
<th>Recursive development</th>
<th>To handle a large group of similar problems</th>
<th>Implementation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structural methodology</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>Long</td>
</tr>
<tr>
<td>Object-Oriented methodology</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>Long</td>
</tr>
<tr>
<td>HDP</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Short</td>
</tr>
</tbody>
</table>

Figure 7.4: Comparisons of different development approaches.
Technically, the logical architecture can map to multiple physical architectures by keeping objects stateless. It can, therefore, easily be scaled up or down, from one machine for low load to as many machines as required to balance the load. In a low-load or development environment, all the functional components can be hosted on a single server. In a high-load environment, the strain on the system can be measured, and each logical tier can be hosted on one or more physical servers. If the Web server is found to be a bottleneck, one or more dedicated Web servers can be added, making possible a round-robin scheme (i.e., continuously repeating sequence, such as the polling of a series of terminals, one after the other, over and over again) that balances the load across each of the servers. If the business logic strains performance, one or more middle-tier servers can be added, and the Web servers can remotely call the middle-tier servers.

Another state-of-the-art technology adopted in this research is WOOM, which is the building block of the implementation process. WOOM is comprised of four logical domains, namely Logic Domain (LD), Presentation Domain (PD), Service Domain (SD), and Database Domain (SD). It is simple to build and easy to modify. More importantly, the WOOM represents both object attributes and relationships between objects in the same data model. This feature enables the conversion between the design and the implementation. In other word, with WOOM, the design is the implementation while the implementation is the design; this characteristic is essential for fast code generation solutions. Moreover, the advantage of the model-driven approach and the possibility of application generation highly reduce the development time on programming, debugging and testing.
In this applied research, object models have been developed using the principles of Booch's OODA methodology (Booch, 1994), which offers four major benefits, namely abstraction, encapsulation, modularity, and hierarchy. First of all, abstraction serves to reduce the complexity of a system by only considering details that are important to the behaviour of an object. The Order Management class is a good example; users can refer to the class without realising it is composed of suppliers, customers and order status. Secondly, the implementations of the ERP objects are hidden, with object interaction being restricted to a small set of well-defined services. In this case, the knowledge base management object is a good example.

Thirdly, Modularity also serves to reduce the apparent complexity of a system, by clustering the entities of similar functionality. As a result the system will consist of a set of loosely coupled components. This can be illustrated by the entities: Sales, Purchaser and Accountant, each of which have been built from more abstract objects in order to fulfill particular roles. Fourthly, hierarchies provide the opportunity for data and function reuse. Hierarchies not only reuse common components, but also reduce the system's apparent complexity, which has aided development of the system.

An object-oriented approach has also made the system's multiple data-types possible. The common base class allows an object to be passed as a parameter without reference to its underlying type - which can be obtained as soon as it becomes significant. Although each data-type is processed differently, these details can be hidden inside each class, together with the functions that allow types to be inter-converted. This is a significant feature, and it separates the problem solving process from the underlying types of data.
From the results of implementation at the collaborated company, it provides rapid and dramatic increases in revenues and profits from improved communications, cross selling, and lead management capabilities. Along with hundreds of thousands in projected annual cost savings, this is expected to produce a seven-figure overall yearly return on investment. In addition, the company plans to take advantage of the system's virtually unlimited scalability by expanding it on a global basis.

In short, this chapter has summarized the research results as well as the benefits and limitations of the proposed methodology. Finally, the next chapter will conclude works and results in this research, and suggest the future development in the related area.
8. CONCLUSIONS AND FUTURE WORK

8.1 A Conclusion Of The Research Work Undertaken

Fueled by advances in information and Internet technology, productivity and supply chain integration have reached new levels. Technology is affordable, abundant, and is redefining the rules of competition. From product conceptualization to manufacturing and marketing, manufacturers are welcoming the active role of customers in every area of business. As the marketplace grows more sophisticated, the push to expand global reach and resources into emerging markets has become stronger than ever.

In view of this globalization trend, this teaching company research provides a new framework for the design and implementation of an enterprise information system for global manufacturing. The results are demonstrated in a proposed HDP and WOOM approach, which expand the global reach of the manufacturers. In the earlier chapters, the author has presented a comprehensive review on global enterprise information system and its design and implementation methodology, and described the application of the proposed HDP model to the design global ERP system under the macroscopic context of business process reengineering. The author further elaborates the novel implementation methodology – WOOM, which provides the essential framework, channel and the interface necessary to build a Web-based ERP system on a global business environment.

The HDP methodology proposed by the author can be summarized as a three-step method to design an enterprise information system for global manufacturing:
Step 1 (conceptualization): To perform a Business Process Reengineering to establish core requirements.

Step 2 (analysis): To execute structural analysis and design to develop a model of the desired behavior.

Step 3 (design and evolution): To construct the object design by using the results of analytical stage to create architecture and evolve the actual implementation.

Basically, the second step is to design a structural ERP model by using IDEF design tool, together with the node index of the model. Note that only those activities closely associated with ERP are considered. The third phase is to model and implement the design by object-oriented technology. Some of programming and database schema illustrations have been included in Appendixes II and III, but detailed discussions have not been included and can be referred to programming knowledge in ASP, C++, Visual Basic, SQL, Java, XML etc. Modern programming technology in OO enables fast and quicker customization of the "As is" ERP model to the "To be" model. This approach combining with BPR provides a powerful and practical customization approach to ERP for different industrial environments.

The WOOM component can be defined as a combination of data and code, which acts upon that data, and together, can be considered as a single unit. The data and code associated to the component defines everything that the component represents (state) and what can be achieved (behavior). The architecture can be logically divided into four major domains;

(i) Logic Domain;
(ii) Presentation Domain;

(iii) Service Domain; and

(iv) Database Domain.

The major technology driver of the decade is the explosive use of the Internet/Intranet/Extranet applications, object technology and client/server architecture. These technologies are embraced in the proposed HDP model. It offers a systematic framework and implementation strategy which provides the essential linkage between business strategy, organization structure with ERP technology under the BPR and continuous improvement context. It enables enterprises whether it is large, medium and small organizations to build an effective information system with a novel approach to gain competitive advantages in an ever-changing global manufacturing environment.

In conclusion, the Hierarchical Design Pyramid (HDP) provides valuable support in the design and development of an ERP system. IDEF0 has been found to be a powerful descriptive tool for depicting the functional and structural relationships, and can be used for functional specifications. It offers a number of features which make it easy to apply and, more importantly, to understand. Furthermore, the HDP together with WOOM provides the integration with OOM that enables a natural thinking way to the implementation of a web-based ERP system. To conclude my discussion, this thesis provides a hierarchical methodology for the design and implementation of an ERP system; the methodology presented here is illustrated with detailed development,
implementation, and results in a company where the author has been employed as teaching company associate.

8.2 Contribution From Teaching Company Scheme

In this research, Teaching Company Scheme is the equivalent to a type of action learning, it is concerned with the developing of new ideas or innovation by placing them into natural experience, seeking to make meaning from experience (Raelin, 1997). The action learning uses a cyclic process to integrate action and learning as proposed by Greenwood et al. (1998). In fact, action learning is increasingly applied in organizations, academia and the community to help understand, implement and effectively manage change (Dick et al., 1999). Dotlich et al. (1998) suggests that the action learning approach provides a situation where the learners become de facto in-house “consultants” for their employers. The employers can benefit greatly from the action learning, which have stood the rigour of academic requirements while solving real problems in the organizations. Unlike external consultants who would disappear after taking the handsome cheques, the action learner (teaching company associate or the author) will stay in the organization and benefit from their pragmatic action learning project, especially in their future career development.

Over the several years of experience in this teaching company research project, the author demonstrated that action learning has a very high applicability to increase learning and overcoming resistance during ERP implementation. The goals of action learning – to provide an empirically based and scientifically sound basis for improving an organization’s practices – would seem consistent with the objective of improving ERP implementation. Indeed, in this teaching company project, it referred
to the design and development of a novel methodology (HDP and WOOM) to successfully implement global enterprise information system.

Whether facilitated by internal or external "researchers" through teaching company scheme, it can contribute valuable knowledge to organizations both engaged in a specific ERP implementation and learning vicariously from the real life experiences. The three stakeholders - company, university, student, would all be benefited from the research. The student has benefited from the practical research experience in a real industrial environment, the company achieves the innovation and productivity, and the university attains a higher level of research scholarly activity and linkage with industry. All in all, in the teaching company project, the research objectives in Section 1.4 have been accomplished as follows:

(i) The importance of information technologies and their associated implementation considerations have been identified and analyzed in Chapter 3;

(ii) Enterprise Resource Planning (ERP) and Business Process Reengineering (BPR) has been integrated by an innovative methodology – HDP – in Chapter 4;

(iii) A framework for the design and implementation of an enterprise information system has been developed by a new model – WOOM – in Chapter 5;

(iv) An architecture of enterprise information system has been designed;

(v) The data model and system have been developed; and
(vi) The HDP and WOOM have been validated in the collaborated company through a case study discussed in Chapter 6.

8.3 Future Works

Two potential areas for further study are recognized and elaborated as follows:

8.3.1 Integration with Real-Time Monitoring System

One of the current major challenges to ERP is to bridge the gap between business logic and production. Manufacturing Execution Systems (MES) uses network computing technology to automate production and process automation. By downloading data from ERP such as Bill-of-material and work schedules, and uploading production performances, it effectively bridges the gap between business planning and shop floor production. In the HDP model described earlier, this can be further improved by using Real-Time Monitoring System (RTMS). A collection of RTMS in worldwide remote production and operation locations enables the development of global manufacturing which forms a closed-loop manufacturing system. The implementation of RTMS is targeted to improve the efficient and management of the shop floor located in dispersed production sites and collected to global business logistics via ERP (Ip, 1998). By the use of data collection devices such as sensors, bar-code readers and Programmable Logic Controllers (PLC), it enables better monitoring of material flow and processes efficiency, and improves production quality (Ng and Ip, 1999).
Schematic diagram of ERP integrated with RTMS is shown in Figure 8.1. According to the production plan, every workstation is monitored by the RTMS when products are being manufactured. Production engineers and planners can monitor the status of assembly lines from the display screen through dynamic graphical simulation function of the monitoring system. All relevant real-time shop floor information will be transferred to the ERP system located in the head office bi-directionally. The production planner and other company staff can easily acquire the real-time information to schedule the production or plan the material consumption. (Stein, 1998).

Figure 8.1: Schematic diagram of ERP integrated with RTMS

If there are any production problems, for example, serious product quality problems or SPC (Statistical Process Control) parameters being abnormal etc., the monitoring
system sends alarm signals to the supervisor so that he can solve the problems quickly. Historical quality test data from all QC stations can also be collected from the real-time database. The supervisor can use the information to predict or estimate the potential production capability of the assembly line with the help of SPC or OLAP. Besides offering the monitoring functions, expert system knowledge-based model can be used for problem diagnosis and correction.

In fact, RTMS today is further enhanced by the application of handheld PCs to provide a wider range of users access critical enterprise data at significantly lower cost (SFA Plus Weekly, 1998). Leading ERP and database developers are bridging their large back-end environments to PalmPilot and Windows CE devices. This trend increases the reach of ERP applications to users ranging from traveling executives to loading-dock workers. Instead of equipping mobile workers with expensive laptops to access home-office data, they will have the option of providing similar access with more price competitive handheld PCs. The main driver for the manufacturers is to involve occasional users who have not been part of an internal business process with an ERP system, which handles all information processing.

On the factory floor, employees could use remote devices to manage the company's information such as quality assurance program, maintenance schedules and production orders. Handheld users will be able to capture in real time of company data or receive alerts when certain condition, such as inventory levels and production breakdowns. The ERP collected to RTMS enables MES to be performed in a global environment where customers, suppliers, engineers, designer, production planners and

172
engineering staff etc. are dispersed in different countries. Therefore, the future research is proposed to do the integration to move to an Intelligent ERP (I-ERP).

### 8.3.2 Web security

In this research, the basic architecture of the whole system is highly related to Internet. However, the security issue of the web-based system has not been fully investigated. If the Internet is to be used to conduct enterprise data communication, mechanisms need to be put in place to protect confidential company information and financial transactions. Lack of security has fostered intranets and firewalls creating segments of infrastructure only partially connected to the Internet. Such segmentation impedes the dissemination of information and offloading of client-side programs.

To implement a highly secure web-based ERP system, the following three areas related to information security mechanisms are needed to further study:

(i) Cryptography

To protect sensitive information such as credit card numbers and pricing information, data must be encrypted and encryption keys provided to authenticated users. Cryptographic security mechanisms need to be applied at the Internet Protocol layer. The amount of security that can be deployed internationally depends on how countries regulate encryption. The W3C Digital Signature Initiative (Joseph, 1999) is addressing the use of digital signature, identity certificate, packing list, and content label technologies to help users decide what code and data to trust on the Web.
(ii) Firewalls

Applications need to cross firewalls. It should be possible for mobile applications to access information inside a firewall where it executes, but not leak that information back out. It should be possible for mobile applications to query information outside a firewall on behalf of a user, but not if the external contacts could leak information retrieved from inside the firewall. Firewalls need to support the encryption of Internet traffic, and provide management and security services to exterior nodes. One short term approach taken by many enterprises is to disable the downloading of Java applets at the firewall by scanning the applets to determine if they are hostile or benign or check if they are in a database of known hostile applets. CORBA IIOP cannot communicate directly across a firewall. To overcome this, HTTP tunneling is used, i.e., IIOP packets are converted to HTTP packets which the firewall recognizes and directs. Mechanisms are needed to enable transparent IIOP communication across firewalls.

(iii) Viruses

Viruses continue to proliferate daily. Documents need to be scanned for viruses when they are downloaded and this is a research area when ERP is totally implemented over Internet.

8.4 Statement Of Originality And Contribution To Knowledge

The major contribution of this research is the hierarchical approach of ERP design and implementation through the HDP methodology. The methodology proposed in this research project can be summarized into two major phases. The first phase is to design a structural ERP model by using the IDEF0 design tool, together with the node
index of the model. Each system, represented in the node index by a number, is broken down into its component parts. Detailed notes are provided with each low-level diagram to outline 'good practice' in each area. Note that only those activities closely associated with ERP are considered here and that, for reasons of confidentiality, only their accompanying discussions are provided. The second phase is to model and implement the design by OOM. The implementation phase in the HDP methodology is not only dependent on the logical design of the ERP conceptual and IDEF design, but also on the implementation of the software system using OOM. The proposed HDP is a design and implementation methodology which overcomes the huge system integration tasks in the development cycle, and minimizes the risk of the development project due to changes in system design and requirement.
REFERENCES


Dotlich, David L. and Noel, James L. (1998), Action learning: how the world’s top companies are re-creating their leaders and themselves. San Francisco, Ca.: Jossey-Bass.


Kurt Salmon Associates (1996), Hong Kong Government Industry Department.


APPENDIX I – QUESTIONNAIRE

Remark: This questionnaire was prepared and distributed through the collaboration company of this project in 3Q 1998. The following is a list of questions in the questionnaire and a summary of the result. Detailed company information and sensitive data is excluded from this thesis.

A) Current Information Technology Management

1. What types of hardware and how many of them are adopted in your firm?
(Tick all that apply)
- Fax Server
- PC Server
- Mainframe
- Remote access server
- Network Modem
- Workstations
- PC
- Others, please specify

2. Which of the following computer applications are implemented in your firm?
(Tick all that apply)
- CAD / CAM / CAE
- Distribution Resources Planning
- MRPII / ERP
- EDI (Electronic Data Interchange)
- Network-WAN, LAM
- Payroll
- Financial management
- Internet / Intranet / Extranet
- Inventory Management
- Order Processing
- Production Scheduling
- Plant Maintenance
- SCADA (Supervisory control and data acquisition)
- Electronic mail
- Others, please specify:

3. Have any shop floor production facilities been integrated into your computer system?
   Yes □  No □
If YES, please indicate the types of production facilities that are connected:  
(Tick all that apply)

☐ CNC machines
☐ Material Transfer System
☐ Inspection/Testing Equipment
☐ PLC
☐ AS/RS
☐ Barcode reader
☐ Automatic Assembly Equipment
☐ Data input pad
☐ Sensors/Actuator
☐ Others, please specify

4. What types of data are frequently transferred between your head office and branch offices either electronically or using traditional methods?  
(Tick all that apply)
☐ Sales Orders
☐ Inventory Control
☐ Material sourcing Records
☐ Engineering Designs
☐ Quality Control Spec. & Records
☐ Production Schedules
☐ Production Records
☐ Delivery Schedules
☐ Marketing Information
☐ Others, please specify

5. Have your company used any computerized Manufacturing Information System?  
Yes ☐ No ☐

If YES, please indicate the areas of applications: (Tick all that apply)
☐ Accounting
☐ Administration
☐ CAD / CAM / CAE
☐ Material Requirement Planning
☐ Product delivery / Shipment
☐ Purchasing Ordering Processing
☐ Quality Assurance & Control
☐ Forecasting
☐ Sales Ordering Processing
☐ Inventory management
☐ Shop Floor Control
☐ Others, please specify

B) Deployment of Global Information Technology

6. Have your firm adopted any internet-based information system?  
Yes ☐ No ☐

If YES, please indicate the areas of applications: (Tick all that apply)
☐ Accounting
☐ Administration
☐ CAD / CAM / CAE
☐ Material Requirement Planning
☐ Quality Assurance & Control
☐ Forecasting
☐ Sales Ordering Processing
☐ Inventory Management
☐ Product delivery / Shipment
☐ Purchasing Ordering Processing
☐ Shop floor Control
☐ Others, please specify

If TICK, please also indicate that are in-house developed: (Tick all that apply)
☐ Accounting
☐ Administration
☐ CAD/CAM / CAE
☐ Material Requirement Planning
☐ Product delivery / Shipment
☐ Purchasing Ordering Processing
☐ Quality Assurance & Control
☐ Forecasting
☐ Sales Ordering Processing
☐ Inventory Management
☐ Shop Floor Control
☐ Others, please specify

7. Do you anticipate that internet-based information system will increasingly be implemented in your firm form a strategy point of view over the next two years?
☐ Yes
☐ No
☐ Don’t know

8. Please rank the order of importance for the following limitations in the application of IT in your firm?
   (1 – most important factor; and 6 – least important factor)
   A lack of capital investment funds
   The application software package do not suit the business requirement
   Rigid program structure make enhancement difficult
   A lack of staff to investigate and manage new technologies
   System review and audit are very complicated and time consuming
   A lack of Top management commitment

9. Please rank the order of importance for the following changes in the application of IT in your firm?
   (1 – most important factor; and 6 – least important factor)
   Review and change the company policy, standards and usual operating procedures
   New software development methodology
   Provide training program to users
   Small and modular computer project development
   Company wide computerization
   Interest group meeting or project team of those involved in computer applications

10. What do you think about the importance of the following essences for improving the efficiency of software design and development cycle?
(1-Very very important, 2-Very important, 3-improtant, 4-Unimportant, 5-Very Unimportant)
Module system architecture 1 2 3 4 5
Design and analysis methodology 1 2 3 4 5
Integration framework 1 2 3 4 5
Use of standard specifications 1 2 3 4 5
C) Background Information

11. Which sector in manufacturing industry can your firm be classified?
☐ Electronics
☐ Toys
☐ Plastics
☐ Metal
☐ Clothing
☐ Others, please specify _________

12. Which manufacturing type can your firm be categorized?
☐ Assembly
☐ Process
☐ Project
☐ Job shop
☐ Others, please specify _________

13. Do your headquarter locate in Hong Kong?
   Yes ☐   No ☐
   If YES, what are the activities that your company conducts in Hong Kong? (Tick all that apply)
   ☐ Production
   ☐ Sales and Marketing
   ☐ Research and Development
   ☐ Logistics
   ☐ Accounting
   ☐ Others, please specify _________

14. How many employees are there in your company including head office, branch offices and production plants? (Tick all that apply)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1-5</th>
<th>6-50</th>
<th>51-100</th>
<th>Over 500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administration, accounting, sales &amp; marketing</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>R&amp;D / Engineering</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Production</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>MIS / EDP</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

15. Where are production plants of your company located in different geographical areas and how many in each location? (Tick all that apply)
☐ Hong Kong
☐ Maintand China
☐ Europe
☐ American
☐ Others
APPENDIX II – SAMPLE SOURCE CODE

The complete set of source code is too large to be printed in here (over 15Mbyte).

The following program is part of the ASP code for the prototype Web-based ERP.

<default.asp>

</default.asp>

<EngMain.asp>

</EngMain.asp>
function chapters.onChangeListener() { 
    if (document.1575212072.itemIdFromOther.value == document.MainForm.itemId.Value) {
        document.1575212072.itemIdFromOther.value = document.MainForm.itemId.Value;
    }
    else {
        if (document.1575212072.itemIdFromOther.value == document.MainForm.itemId.Value) {
            document.1575212072.itemIdFromOther.value = document.MainForm.itemId.Value;
            document.1575212072.itemIdFromOther.value = "Search";
        }
    }
}

</script>
</DIV>
</DIV>

<TABLE border="0" cellPadding="1" cellSpacing="1" width="100">
    <FOOT ID="MainForm" name="MainForm" Method="GetAllAction" Action="MainForm.asp"/>
    <TR>
        <TD FONT face="Arial" color="#000000" style="height:100" width="50"></TD>
        <TD FONT face="Arial" color="#000000" style="height:100" width="50"></TD>
        <TD FONT face="Arial" color="#000000" style="height:100" width="50"></TD>
        <TD FONT face="Arial" color="#000000" style="height:100" width="50"></TD>
        <TD FONT face="Arial" color="#000000" style="height:100" width="50"></TD>
    </TR>
</TABLE>
```html
<select case="idAction" case="Search">
  <%-include file="EngMainItemSearch.asp"%>
  Response.End
</select>

<select case="idAction" case="Search">
  <%-include file="EngMainDescSearch.asp"%>
  Response.End
</select>

<select case="idAction" case="Search">
  <%-include file="EngMainPoSearch.asp"%>
  Response.End
</select>

<select case="idAction" case="Search">
  <%-include file="EngMainPoNew.asp"%>
  Response.End
</select>

<select case="idAction" case="Search">
  <%-include file="EngMainPoEdit.asp"%>
  Response.End
</select>

<select case="idAction" case="Search">
  <%-include file="EngMainPoDelete.asp"%>
  Response.End
</select>
</body>
</html>

<rpt01.asp>

<% @language=Vbscript %>
  Response.Expires = -1
  <%-include file="ASHKpl.asp"%>
  <%-include file="Security.asp"%>
</html>
</HEAD>
<body bgColor="silver" topMargin="0">
<div align="left">

199
```
[Table]

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Color Navy</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>Color Navy</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>Color Navy</td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>Color Navy</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>Color Navy</td>
<td>100</td>
</tr>
</tbody>
</table>

[Search Action] Request: 1dSearchAction

Case Search

CheckSecurity()

SELECT case 1dSearchAction
CASE "Search"

Select * from TABLE1
<td><font face="Courier New" size="1">= "" + rst.Fields(i) + "nbsp" /></font></td>
</tr>
</table>

End Select

</font></td>
</tr>
</table>

function FormatQty(strTemp)
if not IsAmericanTime then
    FormatCurrency = "0"
else
    if SimilarTemp = 0 then
        FormatCurrency = "0"
    else
        FormatCurrency = "1 - " & StrTemp
    end if
end if
```html
<!-- Search -->

```
Else

set objADOConn = Server.CreateObject("ADODB.Connection")
set objADOErr = Server.CreateObject("ADODB.Error")
strSQL = "Select * From TBUser where name = " & strUserName & ""
objADOConn.Open "CрослBD"
objADOConn.BeginTrans
objADOConn.OpenRecordset(strSQL, objADOConn)
objADOConn.Execute strSQL
if objADOConn.Errors.Count = 0 then
objADOConn.CommitTrans
Response.Write "User : " & strUserName & " was added."<BR>
Else
objADOConn.RollbackTrans
Response.Write "User : " & strUserName & " was not added."<BR>
End If

End If

Response.End

case "Edit"

keyUserName = Request("keyUserName")
strSQL = "Select u.name, u.user_group From TBUser u Where " & _
" u.name Like " & keyUserName & ""
set objADOConn = Server.CreateObject("ADODB.Connection")
set objADORecs = Server.CreateObject("ADODB.Recordset")
objADOConn.Open "CрослBD"
objADORecs.Open strSQL, objADOConn
if objADORecs.EOF then
Response.Write "The User doesn't exist."<BR>
Else

<TABLE border=1 cellPadding=1 cellspacing=1 width="100" >
<TR>
<FONT size=1> User Name:</FONT> <INPUT id="UserName" name="UserName" readonly style=HEIGHT: 20px" value="objADORecs("name")" >
</TR>
<TR>
<IF Session("UserGroup") = 0 then

<SELECT id="userGroup" name="userGroup" style=HEIGHT: 20px"

strSQL = "Select * From TBUserGroup"
set objADORecs = Server.CreateObject("ADODB.Recordset")
objADORecs.Open strSQL, objADOConn
Do While Not objADORecs.EOF
ThisSelected = ""
if objADORecs("user_group") = objADORecs("user_group") Then ThisSelected = "selected"
</SELECT>

<OPTION value="objADORecs("user_group")" " " " ThisSelected >> = objADORecs("level")"" >></OPTION>
</IF>

objADORecs.MoveNext

</SELECT>

<TABLE border=1 cellPadding=1 cellspacing=1 width="100" >
<TR>
</TABLE>

<INPUT id="Old Password" style=HEIGHT: 20px" type="password" >
<INPUT id="New Password" style="HEIGHT: 20px" type="password" value=""
</INPUT>
</TD>
</TR>
</TABLE>

ThisIsDisabled = ""
if Session("UserGroup") = 0 then ThisIsDisabled = " disabled"
End If
Response.End
Case "Save"
binCheck = False
strUserName = Request("idUserName")
strOldPwd = Request("idOldPwd")
If Session("UserGroup") <> 0 Then
Set objADOConn = Server.CreateObject("ADODB.Connection")
Set objADORecSQL = Server.CreateObject("ADODB.Recordset")
strSQL = "Select * From TBLUser Where name = "+strUserName+""
And pwd = ""+strOldPwd+""
objADOConn.Open "CoriGlass"
objADORecSQL.Open strSQL, objADOConn
If objADORecSQL.EOF Then
Response.Write "Old Password is not correct."
binCheck = False
End If
objADORecSL.Close
objADOConn.Close
End If
If binCheck Then
strNewPwd = Request("idNewPwd")
strConfPwd = Request("idConfPwd")
If strNewPwd <> strConfPwd Then
Response.Write "New Password is not the same as Confirm Password."
Else
strComma = ""
strSQL = "Update TBLUser Set "
If Session("UserGroup") <> 0 And strNewPwd <> "" Then
Response.Write "Nothing to be changed."
Response.End
End If
If Session("UserGroup") = 0 Then
strComma = ", "
strSQL = "Update TBLUser Set user_Group = "
Request("idUserGroup")
End If
End If
End If
Response.End
Case "Delete"
strUserName = Request("keyUserName")
binCheck = False
If strUserName = "" Then
Response.Write "User Name cannot be NULL."<BR>
binCheck = False
End If
Set objADOConn = Server.CreateObject("ADODB.Connection")
Set objADORecSQL = Server.CreateObject("ADODB.RecordSet")
strSQL = "Select * From TBLUser Where name = " + strUserName + ""
objADOConn.Open "CoriGlass"
objADORecSQL.Open strSQL, objADOConn
If objADORecSQL.EOF Then
Response.Write "User Name does not exist."<BR>
binCheck = False
End If
End If
objADORecSL.Close
objADOConn.Close
End If
If not binCheck Then
Response.Write "User cannot be deleted."<BR>
Else
strSQL = "Delete From TBLUser Where name = " + strUserName + ""

APPENDIX III – DATABASE SCHEMA

<table>
<thead>
<tr>
<th>Table</th>
<th>Field</th>
<th>Type</th>
<th>Len</th>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBLUser</td>
<td>name</td>
<td>Text</td>
<td>16K</td>
<td></td>
<td>User Name</td>
</tr>
<tr>
<td></td>
<td>pwd</td>
<td>Text</td>
<td>16</td>
<td></td>
<td>User Password</td>
</tr>
<tr>
<td></td>
<td>user_group</td>
<td>Num</td>
<td></td>
<td></td>
<td>User Group; TBLUserGroup.user_group</td>
</tr>
<tr>
<td>TBLUserGroup</td>
<td>user_group</td>
<td>Num</td>
<td>K</td>
<td></td>
<td>12—&gt;HK User (ReadOnly)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18—&gt;UK User (Readonly)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>32—&gt;HK User (Full Access)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>38—&gt;UK User (Full Access)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>50—&gt;Administrator</td>
</tr>
<tr>
<td></td>
<td>level</td>
<td>Text</td>
<td>32</td>
<td></td>
<td>Level Description</td>
</tr>
<tr>
<td>TBLVendor</td>
<td>code</td>
<td>Text</td>
<td>5K</td>
<td></td>
<td>Vendor Code</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>Text</td>
<td>50</td>
<td></td>
<td>Vendor Name</td>
</tr>
<tr>
<td>TBLContractor</td>
<td>code</td>
<td>Num</td>
<td>K</td>
<td></td>
<td>Contractor Code</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>Text</td>
<td>50</td>
<td></td>
<td>Contractor Name</td>
</tr>
<tr>
<td>TBLEngineer</td>
<td>code</td>
<td>Text</td>
<td>2K</td>
<td></td>
<td>Initial</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>Text</td>
<td>50</td>
<td></td>
<td>Full Name</td>
</tr>
<tr>
<td>TBLProductCat</td>
<td>code</td>
<td>Text</td>
<td>5K</td>
<td></td>
<td>Cat Code</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>50</td>
<td></td>
<td>Cat Description</td>
</tr>
<tr>
<td>TBLCert</td>
<td>code</td>
<td>Num</td>
<td>K</td>
<td></td>
<td>Cert Code</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>32</td>
<td></td>
<td>Cert. Description; (&quot;Non-limited Cert.&quot;, &quot;Limited-Cert.&quot;, etc)</td>
</tr>
<tr>
<td>TBLPack</td>
<td>code</td>
<td>Num</td>
<td>K</td>
<td></td>
<td>Packing Code</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>32</td>
<td></td>
<td>Packing Description</td>
</tr>
<tr>
<td>TBLMarket</td>
<td>code</td>
<td>Num</td>
<td>K</td>
<td></td>
<td>Market Code</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>32</td>
<td></td>
<td>Market Name</td>
</tr>
<tr>
<td>TBLItem</td>
<td>model</td>
<td>Text</td>
<td>5K</td>
<td></td>
<td>Model Number</td>
</tr>
<tr>
<td></td>
<td>cat</td>
<td>Text</td>
<td>5</td>
<td></td>
<td>Category; TBLProductCat.code</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>50</td>
<td></td>
<td>Description</td>
</tr>
<tr>
<td></td>
<td>cert</td>
<td>Num</td>
<td></td>
<td></td>
<td>Cert; TBLCert.code</td>
</tr>
<tr>
<td></td>
<td>rel_month</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>rel_year</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>target_price</td>
<td>Num</td>
<td></td>
<td></td>
<td>Target Price</td>
</tr>
<tr>
<td></td>
<td>Le_cert</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cert_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uk_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>export_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fob_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mass_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>states_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>premium_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>other_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_uk_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_export_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_fob_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_direct_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_mass_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_states_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_premium_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new_other_qty</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>approval</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>change</td>
<td>Num</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TBLPo**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Po</td>
<td>Text</td>
<td>16K P.O. Number</td>
</tr>
<tr>
<td>Po_rev</td>
<td>Num</td>
<td>P.O. Revisied Number</td>
</tr>
<tr>
<td>model</td>
<td>Text</td>
<td>5 Model Number: TBLitem.model</td>
</tr>
<tr>
<td>active</td>
<td>Num</td>
<td>Active quotation? (YN)</td>
</tr>
<tr>
<td>rel_month</td>
<td>Num</td>
<td>Release Month</td>
</tr>
<tr>
<td>rel_year</td>
<td>Num</td>
<td>Release Year</td>
</tr>
<tr>
<td>engineer</td>
<td>Text</td>
<td>2 Engineer: TBLEngineer.code</td>
</tr>
<tr>
<td>last_update_date</td>
<td>Date/Time</td>
<td>Last Update Date</td>
</tr>
<tr>
<td>pic_path</td>
<td>Text</td>
<td>200 Picture Path</td>
</tr>
<tr>
<td>unit_price</td>
<td>Num</td>
<td>Unit Price</td>
</tr>
<tr>
<td>target_price</td>
<td>Num</td>
<td>Target Price, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>uk_qty</td>
<td>Num</td>
<td>UK Qty, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>export_qty</td>
<td>Num</td>
<td>Export Qty, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>direct_qty</td>
<td>Num</td>
<td>Direct Qty, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>mass_qty</td>
<td>Num</td>
<td>Mass Qty, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>fob_qty</td>
<td>Num</td>
<td>FOB Qty, Not Use, use field in tblitem instead</td>
</tr>
<tr>
<td>cert_qty</td>
<td>Num</td>
<td>Cert Qty</td>
</tr>
<tr>
<td>le_cert</td>
<td>Num</td>
<td>LE Cert. Production Qty</td>
</tr>
<tr>
<td>prod_qty</td>
<td>Num</td>
<td>Production Qty (Total Qty)</td>
</tr>
<tr>
<td>insp_date</td>
<td>Date/Time</td>
<td>Inspection Date</td>
</tr>
<tr>
<td>insp_date</td>
<td>Date/Time</td>
<td>Revised Inspection Date</td>
</tr>
<tr>
<td>ven_insp_date</td>
<td>Date/Time</td>
<td>Vendor Inspection Date</td>
</tr>
<tr>
<td>ven_insp_date</td>
<td>Date/Time</td>
<td>Revised Vendor Inspection Date</td>
</tr>
<tr>
<td>uk_ref_date</td>
<td>Date/Time</td>
<td>UK Reference Date</td>
</tr>
<tr>
<td>ven_ref_date</td>
<td>Date/Time</td>
<td>Vendor Reference Date</td>
</tr>
<tr>
<td>plate_l_qty</td>
<td>Num</td>
<td>Tampo Plate (Large) Qty</td>
</tr>
<tr>
<td>plate_m_qty</td>
<td>Num</td>
<td>Tampo Plate (Medium) Qty</td>
</tr>
<tr>
<td>plate_s_qty</td>
<td>Num</td>
<td>Tampo Plate (Small) Qty</td>
</tr>
<tr>
<td>plate_l_price</td>
<td>Num</td>
<td>Tampo Plate (Large) Price</td>
</tr>
<tr>
<td>plate_m_price</td>
<td>Num</td>
<td>Tampo Plate (Medium) Price</td>
</tr>
<tr>
<td>plate_s_price</td>
<td>Num</td>
<td>Tampo Plate (Small) Price</td>
</tr>
<tr>
<td>book_mask_qty</td>
<td>Num</td>
<td>Book Mask Qty</td>
</tr>
<tr>
<td>plate_mask_qty</td>
<td>Num</td>
<td>Plate Mask Qty</td>
</tr>
<tr>
<td>book_mask_price</td>
<td>Num</td>
<td>Book Mask Price</td>
</tr>
<tr>
<td>plate_mask_price</td>
<td>Num</td>
<td>Plate Mask Price</td>
</tr>
<tr>
<td>test_no</td>
<td>Text</td>
<td>16 Lab Test Number</td>
</tr>
<tr>
<td>test_qty</td>
<td>Num</td>
<td>16 Inspected Qty</td>
</tr>
<tr>
<td>test_date</td>
<td>Date/Time</td>
<td>Lab Test Date</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>---------------</td>
</tr>
<tr>
<td>std_pack</td>
<td>Num</td>
<td>Standard Pack? (Y/N)</td>
</tr>
<tr>
<td>pack_type</td>
<td>Num</td>
<td>Pack Type; TBLPack.code</td>
</tr>
<tr>
<td>mast_snd_u0k</td>
<td>Date/Time</td>
<td>Decorated Master Sent from UK</td>
</tr>
<tr>
<td>mast_rcv_u0k</td>
<td>Date/Time</td>
<td>Decorated Master Received from UK</td>
</tr>
<tr>
<td>mast_snd_v0en</td>
<td>Date/Time</td>
<td>Decorated Master Sent to Vendor</td>
</tr>
<tr>
<td>dec_art_snd_u0k</td>
<td>Date/Time</td>
<td>Decorated Master Artwork/Disk sent from UK</td>
</tr>
<tr>
<td>dec_art_rcv_u0k</td>
<td>Date/Time</td>
<td>Decorated Master Artwork/Disk Received from UK</td>
</tr>
<tr>
<td>dec_art_snd_v0en</td>
<td>Date/Time</td>
<td>Decorated Master Artwork/Disk Sent to Vendor</td>
</tr>
<tr>
<td>pck_art_snd_u0k</td>
<td>Date/Time</td>
<td>Packing Artwork/Disk Sent from UK</td>
</tr>
<tr>
<td>pck_art_rcv_u0k</td>
<td>Date/Time</td>
<td>Packing Artwork/Disk Received from UK</td>
</tr>
<tr>
<td>pck_art_snd_v0en</td>
<td>Date/Time</td>
<td>Packing Artwork/Disk Sent to Vendor</td>
</tr>
<tr>
<td>Pp_rcv_v0en</td>
<td>Date/Time</td>
<td>Pre-production Sample Received from Vendor</td>
</tr>
<tr>
<td>Pp_snd_u0k</td>
<td>Date/Time</td>
<td>Pre-production Sample Sent to UK for Comment</td>
</tr>
<tr>
<td>Pp_rcv_u0k</td>
<td>Date/Time</td>
<td>Pre-production Sample Received to UK for Comment</td>
</tr>
<tr>
<td>Pp_cmt_snd_u0k</td>
<td>Date/Time</td>
<td>Comment of Pre-production Sample Sent to UK</td>
</tr>
<tr>
<td>Pp_cmt_rcv_u0k</td>
<td>Date/Time</td>
<td>Comment of Pre-production Sample Received from UK</td>
</tr>
<tr>
<td>Pp_cmt_snd_v0en</td>
<td>Date/Time</td>
<td>Comment of Pre-production Sample Sent to Vendor</td>
</tr>
<tr>
<td>Pp_approval</td>
<td>Date/Time</td>
<td></td>
</tr>
<tr>
<td>Pm_rcv_v0en</td>
<td>Date/Time</td>
<td>Print Proof Received from Vendor</td>
</tr>
<tr>
<td>Pm_snd_u0k</td>
<td>Date/Time</td>
<td>Print Proof Sent to UK</td>
</tr>
<tr>
<td>Pm_rcv_u0k</td>
<td>Date/Time</td>
<td>Print Proof Received to UK</td>
</tr>
<tr>
<td>Pm_cmt_snd_u0k</td>
<td>Date/Time</td>
<td>Print Proof Comment Sent from UK</td>
</tr>
<tr>
<td>Pm_cmt_rcv_u0k</td>
<td>Date/Time</td>
<td>Print Proof Comment Received from UK</td>
</tr>
<tr>
<td>Pm_cmt_snd_v0en</td>
<td>Date/Time</td>
<td>Print Proof Comment Sent to Vendor</td>
</tr>
<tr>
<td>note</td>
<td>Text</td>
<td>100</td>
</tr>
<tr>
<td>note</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>obs_date</td>
<td>Date/Time</td>
<td>OBS Date</td>
</tr>
<tr>
<td>Req_date</td>
<td>Date/Time</td>
<td>Request Date</td>
</tr>
<tr>
<td>vendor1</td>
<td>Text</td>
<td>5</td>
</tr>
<tr>
<td>vendor2</td>
<td>Text</td>
<td>5</td>
</tr>
<tr>
<td>vendor3</td>
<td>Text</td>
<td>5</td>
</tr>
<tr>
<td>contractor1</td>
<td>Num</td>
<td>SubContractor #1; TBLContractor.code</td>
</tr>
<tr>
<td>contractor2</td>
<td>Num</td>
<td>SubContractor #2; TBLContractor.code</td>
</tr>
<tr>
<td>contractor3</td>
<td>Num</td>
<td>SubContractor #3; TBLContractor.code</td>
</tr>
<tr>
<td>accessory1</td>
<td>Text</td>
<td>70</td>
</tr>
<tr>
<td>accessory2</td>
<td>Text</td>
<td>70</td>
</tr>
<tr>
<td>accessory3</td>
<td>Text</td>
<td>70</td>
</tr>
<tr>
<td>act_rcv_v0en</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>act_snd_u0k</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>adv_snd_u0k</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>adv_sub_v0en</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>cmt_snd_v0en</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>cmt_snd_u0k</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>cmt_UK_rep</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>bud_no</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>pck_cost_inv_no</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>Hk_pck_cost</td>
<td>Num</td>
<td>hk packing cost</td>
</tr>
<tr>
<td>Uk_pck_cost</td>
<td>Num</td>
<td>uk packing cost</td>
</tr>
<tr>
<td>misc</td>
<td>Num</td>
<td>misc</td>
</tr>
<tr>
<td>Line_tool_inv_no</td>
<td>Text</td>
<td>50</td>
</tr>
<tr>
<td>Line_tool_bud</td>
<td>Num</td>
<td>line tool budget</td>
</tr>
<tr>
<td>Lab_test_bud</td>
<td>Num</td>
<td>lab test budget</td>
</tr>
<tr>
<td>Lab_test_cha</td>
<td>Num</td>
<td>lab test charge</td>
</tr>
<tr>
<td>remark</td>
<td>Text</td>
<td>100</td>
</tr>
<tr>
<td>creation_date</td>
<td>Date/Time</td>
<td>creation date</td>
</tr>
<tr>
<td>unit_price_vendor1</td>
<td>Num</td>
<td>Unit Price for vendor</td>
</tr>
<tr>
<td>unit_price_vendor2</td>
<td>Num</td>
<td>Unit Price for vendor</td>
</tr>
<tr>
<td>unit_price_vendor3</td>
<td>Num</td>
<td>Unit Price for vendor</td>
</tr>
<tr>
<td>TBLSi</td>
<td>code</td>
<td>Text</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>Po</td>
<td>Text</td>
</tr>
<tr>
<td></td>
<td>market</td>
<td>Num</td>
</tr>
<tr>
<td></td>
<td>si_qty</td>
<td>Num</td>
</tr>
<tr>
<td></td>
<td>shp_date</td>
<td>Date/Time</td>
</tr>
<tr>
<td></td>
<td>shp_qty</td>
<td>Num</td>
</tr>
<tr>
<td></td>
<td>si_rev</td>
<td>Num</td>
</tr>
<tr>
<td></td>
<td>last_update_date</td>
<td>Date/Time</td>
</tr>
<tr>
<td></td>
<td>creation_date</td>
<td>Date/Time</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TBLTool</th>
<th>code</th>
<th>Text</th>
<th>10K</th>
<th>Tool Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>model</td>
<td>Text</td>
<td>5</td>
<td>TBLItem.model</td>
</tr>
<tr>
<td></td>
<td>last_transfer</td>
<td>Date/Time</td>
<td></td>
<td>Last Transfer Date</td>
</tr>
<tr>
<td></td>
<td>last_audit</td>
<td>Date/Time</td>
<td></td>
<td>Last Audit Date</td>
</tr>
<tr>
<td></td>
<td>from_loc</td>
<td>Text</td>
<td>50</td>
<td>From Location</td>
</tr>
<tr>
<td></td>
<td>To_loc</td>
<td>Text</td>
<td>50</td>
<td>To Location</td>
</tr>
<tr>
<td></td>
<td>position</td>
<td>Text</td>
<td>3</td>
<td>Position</td>
</tr>
<tr>
<td></td>
<td>owner</td>
<td>Text</td>
<td></td>
<td>Owner</td>
</tr>
<tr>
<td></td>
<td>manufacturer</td>
<td>Text</td>
<td>50</td>
<td>Manufacturer</td>
</tr>
<tr>
<td></td>
<td>machine_type</td>
<td>Text</td>
<td>8</td>
<td>Machine Type</td>
</tr>
<tr>
<td></td>
<td>cost</td>
<td>Num</td>
<td></td>
<td>Cost</td>
</tr>
<tr>
<td></td>
<td>remark</td>
<td>Text</td>
<td>100</td>
<td>remark</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TBLPart</th>
<th>code</th>
<th>Text</th>
<th>10K</th>
<th>Part Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>tool</td>
<td>Text</td>
<td>10</td>
<td>Tool Number; TBLTool.code</td>
</tr>
<tr>
<td></td>
<td>qty</td>
<td>Num</td>
<td></td>
<td>Part Qty</td>
</tr>
<tr>
<td></td>
<td>material</td>
<td>Text</td>
<td>50</td>
<td>Part Material</td>
</tr>
<tr>
<td></td>
<td>desc</td>
<td>Text</td>
<td>50</td>
<td>Description</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TBLLog</th>
<th>ID</th>
<th>Num</th>
<th></th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>update_date</td>
<td>Date/Time</td>
<td></td>
<td>update date</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>Text</td>
<td>16</td>
<td>update by</td>
</tr>
<tr>
<td></td>
<td>Type</td>
<td>Text</td>
<td>16</td>
<td>Log Type</td>
</tr>
<tr>
<td></td>
<td>itemPoName</td>
<td>Text</td>
<td>10</td>
<td>po number</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Text</td>
<td></td>
<td>Action</td>
</tr>
</tbody>
</table>